IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

BRADIUM TECHNOLOGIES LLC, *Plaintiff,* v. MICROSOFT CORPORATION, *Defendant.*	**Case No. 1:15-cv-00031-RGA** **JURY TRIAL DEMANDED**

FIRST AMENDED COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff Bradium Technologies LLC ("Bradium") alleges as follows against defendant Microsoft Corporation ("Microsoft"):

THE PARTIES

1. Bradium Technologies LLC is a Delaware Limited Liability Company (LLC) with a place of business at 75 Montebello Road, Suffern, New York 10901-3746.

2. Microsoft Corporation is a Washington corporation with a place of business at One Microsoft Way, Redmond, WA, 98052-6399. Microsoft may be served via service upon Corporation Service Company, 2711 Centerville Rd., Suite 400, Wilmington, DE 19808.

JURISDICTION AND VENUE

3. This is an action for patent infringement under the patent laws of the United States, Title 35 of the United States Code, arising from Microsoft's making, using, selling, putting into service, and/or offering for sale in the United States, and/or importing into the

United States, and/or supplying in or from the United States, various products, services, and components, including those related to Bing Maps, and/or inducing others to do the same, and/or contributing to others doing the same, and/or inducing or contributing to others combining such components in an infringing manner, during the term of U.S. Patent Nos. 7,139,794, 7,908,343, 8,924,506, and 9,253,239, and from the date Microsoft had notice of the application published as United States Patent Application Publication No. 2011/0175914. This Court has jurisdiction over the subject matter of this action pursuant to 28 U.S.C. §§ 1331 and 1338(a) because this action arises under the patent laws of the United States, including at least 35 U.S.C. §§ 154(d), 271(a), (b), (c), and (f).

4. This Court has personal jurisdiction over Microsoft because, among other things, Microsoft has made, used, sold, and/or offered for sale Bing Maps products and services in the State of Delaware and within this District, and the causes of action alleged herein arise in part from such conduct, and because Microsoft regularly and systematically transacts business in this District at least through its store located at 137 Christiana Mall, Newark, DE 19702.

5. Furthermore, Microsoft has purposefully availed itself of the benefits of doing business in the State of Delaware and in this District by, among other things, the acts alleged in Paragraph 4 of this Complaint and the acts of filing numerous lawsuits in this District, including, for example, *Microsoft Corp. v. Robocast Inc.*, C.A. 13-cv-313 (D. Del. Feb. 25, 2013), D.I. 1 (Microsoft complaint alleging patent infringement); *Microsoft Corp., et al. v. GeoTag Inc.*, C.A. 11-cv-175 (D Del. Mar. 1, 2011), D.I. 1 (Microsoft complaint seeking declaratory judgment of patent invalidity). Microsoft has not contested that this District is a proper venue and that it is subject to personal jurisdiction in this venue in past litigation. *Interdigital Commc'ns, et al. v.*

Nokia Corp., *et al.*, C.A. 13-cv-010 (D Del. Mar 7, 2013), D.I. 14 at ¶¶ 8-9 (Microsoft answer, consenting to litigate patent infringement matter in the District of Delaware).

6. The exercise of personal jurisdiction over Microsoft would not offend traditional notions of fair play and substantial justice.

7. Venue is proper in this District pursuant to 28 U.S.C. §§ 1391 and 1400 because Microsoft resides or is deemed to reside in this District, is subject to personal jurisdiction in this District, has committed acts of infringement in this District, has a regular and established place of business in this District, may be found in this District, and has one or more agents who reside in or may be found in this District.

BACKGROUND

8. This lawsuit asserts causes of action for infringement of United States Patent Nos. 7,139,794 (the "'794 patent"), 7,908,343 (the "'343 patent"), 8,924,506 (the "'506 patent"), 9,253,239 (the "'239 patent"), and of United States Patent Application Publication No. 2011/0175914 ("Publication No. 2011/0175914") (collectively, the "Asserted Patents").

9. Isaac Levanon and Yoni Lavi invented the technology claimed in the '794, '343, '506 and '239 patents and Publication No. 2011/0175914. A company, 3DVU, was created in order to commercialize this technology.

10. 3DVU met with Microsoft executives on multiple occasions in or around 2005, including a meeting at Microsoft in about September 2005.

11. At these meetings, 3DVU and Microsoft discussed the possible acquisition by Microsoft of the technology invented by Messrs. Levanon and Lavi. In relation to these

meetings, Microsoft specifically requested information about, and 3DVU disclosed to Microsoft executives and engineers involved in mapping technology at Microsoft, the technology and the patent applications that ultimately led to the ’794, ’343, ’506 and ’239 patents and Publication No. 2011/0175914. 3DVU demonstrated a prototype for Microsoft personnel.

12. As part of the acquisition discussions, and, on information and belief, to induce 3DVU to continue making further disclosures regarding technology and other matters to Microsoft, Microsoft executives informed 3DVU that they had “strategic approval” from Microsoft to proceed with the acquisition deal.

13. Instead of acquiring 3DVU, or purchasing or seeking to license this technology, upon information and belief, Microsoft without consent or authorization began to incorporate the technology of the ’794, ’343, ’506, and ’239 patents and Publication No. 2011/0175914 into its products and services.

14. Microsoft has prior knowledge of at least the ’794 and ’343 patents as demonstrated by the fact that its own patents and patent applications refer to the ’794 and ’343 patents. The ’794 patent was cited during the prosecution of U.S. Patent No. 7,664,870 (“the ’870 patent”) on August 15, 2008. The ’870 patent was assigned to Microsoft as of August 15, 2008. The ’794 patent was also cited during the prosecution of U.S. Patent No. 8,386,560 (“the ’560 patent”) on September 8, 2008. The ’560 patent was assigned to Microsoft as of September 8, 2008. The ’343 patent, and its Publication No. 2010/0064002, were cited in an International Search Report for International Application No. PCT/US2011/038008, for which Microsoft was the applicant, on December 28, 2011. Publication No. 2010/0064002 is listed on

the face of U.S. Patent No. 8,446,441 that issued from International Application No. PCT/US2011/038008.

15. Bradium and its managing member and licensing agent General Patent Corporation wrote to Microsoft on May 27, 2014, and again on December 23, 2014, regarding the patent family that includes the ’794, ’343, ’506, and ’239 patents, but, despite these invitations, Microsoft did not enter into licensing or business discussions with Bradium or General Patent Corporation. (Exhibit A.) In both the May 27, 2014 and December 23, 2014 letters, Bradium and General Patent Corporation noted the ’794 and ’343 patents, and informed Microsoft of Application No. 13/027,929, which issued as the ’506 patent on December 30, 2014, and which published as Publication No. 2011/0175914 on July 21, 2011.

16. Bradium wrote to Microsoft on February 2, 2016 informing Microsoft of the issuance of the ’239 patent. A true and correct copy of the February 2, 2016 letter is attached hereto as Exhibit B. On information and belief, Microsoft was already aware prior to February 2, 2016 that the application for the ’239 patent had been granted and that the patent would issue. In the February 2, 2016 letter, Bradium informed Microsoft that Microsoft was directly and indirectly infringing the ’239 patent by, among other things, using, putting into service and inducing users to use Bing Maps, Bing Search and Bing Maps Preview products and services. Bradium requested that Microsoft immediately cease its infringing activities regarding the ’239 patent, and that Microsoft inform Bradium by February 16, 2016 whether Microsoft would comply. Microsoft responded by letter dated February 16, 2016, in which letter Microsoft failed to state or indicate in any way that it would comply with Bradium’s request. Bradium sent a follow up letter dated February 17, 2016 noting Microsoft’s failure to state whether it would comply with Bradium’s request and stating that Bradium would assume based upon such failure

that Microsoft did not intend to comply. On information and belief, Microsoft has not in fact complied with Bradium's request, and Microsoft's conduct in continuing its infringing conduct with respect to the '239 patent by, among other things, using, putting into service and inducing users to use Bing Maps, Bing Search and Bing Maps Preview products and services after receiving notice of infringement of such patents, is objectively reckless and not in good faith.

17. Microsoft provides mapping products and services, including Bing Maps products and services, in the United States, including in the District of Delaware. Microsoft distributes at least its Bing Search and Bing Maps Preview applications in this District, and Microsoft makes Bing Maps available via the web in this District.

18. At least Microsoft's currently-available mapping products and services, including Bing Maps products and services, use the patented technology of the '794, '343, '506, and '239 patents and Publication No. 2011/0175914.

19. Bing Maps has been a part of Microsoft's Online Services Division ("OSD"), including during Microsoft's fiscal year 2013. The sale of search and display advertising accounted for nearly all of OSD's revenue in fiscal year 2013.

20. Bing Maps generates revenue for Microsoft directly and/or indirectly through the sale of search and display advertising and/or licensing.

21. OSD's online advertising revenue for fiscal year 2013 was approximately $3.0 billion dollars.

22. Microsoft has distributed, and continues to distribute, the Bing Maps Preview application via the Windows store, and the Bing Search application (which includes Bing Maps)

via at least the Google Play Store, the iOS App Store, and the Windows Store, and Microsoft has made, for example, Bing Maps available via Microsoft web sites to users via at least smartphone web browsers.

23. Microsoft advertises and promotes mapping products and services, including Bing Maps and Bing Maps Preview. The Google Play Store indicates that the Bing Search application (which includes Bing Maps) is offered by Microsoft, and that the Bing Search application has been downloaded over one million times. The iOS App Store indicates that Microsoft is the seller of the Bing Search application. The Windows Store indicates that the Bing Maps Preview application is offered by Microsoft.

24. Microsoft provides instructions for users of its mapping products and services, including Bing Maps and Bing Maps Preview, via, for example, Microsoft's website, via Bing Blogs, and via the Bing Search application description that is displayed, for example, on the iOS App Store. For example, the description for Bing Search that is displayed on the iOS App Store instructs users, under the "Maps" heading, to "[g]et walking, driving, and transit directions with current traffic conditions," and to "[e]xplore shops, restaurants and other places with ratings, prices, and other detailed information." As another example, Microsoft provides instructions for users of the Bing Maps Preview application for Windows 8.1 in a September 10, 2014 Bing Blogs posting entitled "Get Around Town Faster with Bing Maps Preview App," http://blogs.bing.com/search/2014/09/10/get-around-town-faster-with-bing-maps-preview-app/, and via a YouTube video entitled "Bing Maps Preview App," that was posted by Microsoft's Bing Maps Team on December 5, 2013, http://www.youtube.com/watch?v=6X5a2Wj4URM.

25. The Microsoft Bing Maps team makes some test data related to Bing Maps publicly available for Bing Maps. *See* Ricky Brundritt, *Location Intelligence for Windows Store Apps* 199 (2014), *available at* http://blogs.msdn.com/b/rbrundritt/archive/2014/03/04/free-ebook-location-intelligence-for-windows-store-apps.aspx.

26. On information and belief, Microsoft also offers training on the use of at least the Bing Search application and Bing Maps, as part of the on-site training at one or more Microsoft Stores, including as part of the "Windows 8.1: Fundamentals of The New Windows" course that is offered at the Microsoft Store in Newark, Delaware. The "event details" for this course indicate that the event includes instruction on apps. *See Newark Event Calendar*, Microsoft Store, http://www.microsoft.com/en-us/store/locations/de/newark/christiana-mall/store-16#events (last visited January 5, 2014).

27. On June 16, 2015, Microsoft petitioned for *inter partes* review of all claims of the '506, '343, and '794 patents on grounds of obviousness, in IPR Nos. IPR2015-01434, IPR2015-01435, and IPR2015-01432, respectively.

28. On December 23, 2015, the Patent Trial and Appeal Board denied institution of *inter partes* review of the '343 and '506 patents, rejecting each obviousness challenge asserted by Microsoft in its petition and concluding that Microsoft had failed to demonstrate a reasonable likelihood of success in proving that even a single claim of those patents was invalid. In particular, the Patent Trial and Appeal Board rejected Microsoft's petition regarding the '343 patent that was based on arguments that the following combinations of references invalidated the claims: Michael Potmesil, *Maps Alive: Viewing Geospatial Information on the WWW*, Computer Networks and ISDN Systems Vol. 29, No. 7 (Aug. 1997) ("*Potmesil*") in view of PCT Patent

Publication WO 99/41675 ("*Hornbacker*"); *Potmesil* in view of *Hornbacker* and Peter Lindstrom *et al.*, *An Integrated Global GIS and Visual Simulation System*, Graphics, Visualization & Usability Center, Georgia Institute of Technology, undated ("*Lindstrom*"); U.S. Patent No. 6,650,998 ("*Rutledge*") in view of U.S. Patent No. 5,682,441 ("*Ligtenberg*") and U.S. Patent No. 6,118,456 ("*Cooper*"); *Rutledge* in view of *Ligtenberg*, *Cooper* and U.S Patent No. 5,940,117 ("*Hassan*"); and *Rutledge* in view of *Ligtenberg*, *Cooper* and PCT Publication No. WO 98/15920. *See* IPR2015-01434, Paper 15. The Patent Trial and Appeal Board also rejected Microsoft's petition regarding the '506 patent that was based on arguments that the following combinations of references invalidated the claims: *Potmesil* in view of *Hornbacker* and *Lindstrom*; *Rutledge* in view of *Ligtenberg* and *Cooper*; and *Rutledge* in view of *Ligtenberg*, *Cooper* and *Hassan*. *See* IPR2015-01435, Paper 14. Additionally, the Patent Trial and Appeal Board denied one of the grounds for Microsoft's challenges to the claims of the '794 patent.

COUNT I
INFRINGEMENT OF U.S. PATENT NO. 7,139,794

29. The contents of Paragraphs 1 through 28 are incorporated by reference as if specifically set forth herein.

30. On November 21, 2006, United States Letters Patent No. 7,139,794 for "System and Methods for Network Image Delivery with Dynamic Viewing Frustum Optimized for Limited Bandwidth Communication Channels," was duly and legally issued to Isaac Levanon and Yoni Lavi. A Certificate of Correction issued for the '794 patent on July 1, 2014. All rights and interest in the '794 patent have been assigned to Bradium, including the right to sue for past damages. A true and correct copy of the '794 patent, including the Certificate of Correction, is attached hereto as Exhibit C.

31. On information and belief, Microsoft has directly infringed and continues to directly infringe one or more claims of the '794 patent, including at least claim 2, as the claims are properly construed both prior to and after the issuance of the Certificate of Correction. On information and belief, the infringing acts include performing each step of claim 2 by or under the direction and control of Microsoft, including at least using one or more of: (i) Microsoft's Bing Search application for Android on a computer device such as a smartphone; (ii) Microsoft's Bing Search application for iOS on a computer device such as a smartphone; (iii) Microsoft's Bing Maps Preview application for Microsoft Windows on a computer device such as a smartphone; and/or (iv) Microsoft's Bing Maps web sites available via smartphone web browsers on a computer device such as a smartphone.

32. For example, on information and belief, Microsoft's Bing Search application for Android on a computer device such as a smartphone, determines a viewpoint orientation with respect to a Bing Maps image displayed within a three-dimensional space, in response to user navigational commands when employing the "bird's eye" map style.

33. Further, on information and belief, Microsoft's Bing Search application for Android on a computer device such as a smartphone, in combination with Microsoft servers, requests Bing Maps map tiles in a priority order to provide progressive resolution enhancement of the Bing Maps image.

34. On information and belief, Bing Maps map tiles are image parcels that correspond to a region of the map image and that have an associated resolution. *See, e.g.*, Joe Schwartz, *Bing Maps Tile System*, Microsoft Developer Network, http://msdn.microsoft.com/en-

us/library/bb259689.aspx (last visited Oct. 31, 2014), which includes the following graphic that describes Bing Maps map tiles:

35. Further, on information and belief, Microsoft's Bing Search application for Android on a computer device such as a smartphone, in combination with Microsoft servers, stores and renders the Bing Maps map tiles, and the priority of the map tiles is reevaluated in response to user navigational commands.

36. Microsoft is therefore liable for direct infringement of the '794 patent pursuant to 35 U.S.C. § 271(a).

37. As of the date of the Complaint (D.I. 1), Microsoft is inducing infringement of at least claim 2 of the '794 patent under 35 U.S.C. § 271(b), as the claims are properly construed both prior to and after the issuance of the Certificate of Correction, because Microsoft has intended, and continues to intend, to cause end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner covered by one or more claims of the '794 patent, including claim 2.

38. Microsoft instructs and encourages end users to use at least Bing Maps and the Bing Search application in a manner covered by one or more claims of the '794 patent, including claim 2.

39. On information and belief, end users have used, and continue to use, at least, Bing Search, Bing Maps Preview, and Bing Maps in an infringing manner, as the claims are properly construed both prior to and after the issuance of the Certificate of Correction.

40. As of the date of the Complaint (D.I. 1), Microsoft is actively inducing and encouraging direct infringement by end users by offering users Bing Rewards for using Microsoft and/or Bing products and services, including the Bing Search application. Microsoft also provides instructions to encourage end users to use at least Bing Maps, Bing Maps Preview and the Bing Search application in a manner covered by one or more claims of the '794 patent, including claim 2, as the claims are properly construed both prior to and after the issuance of the Certificate of Correction.

41. On information and belief, Microsoft as of the date of the Complaint (D.I. 1) had and has knowledge of the '794 patent and knowledge that end users' use of at least Bing Maps, Bing Maps Preview, and the Bing Search application infringes the '794 patent. On information and belief, as of the date of the Complaint (D.I. 1) Microsoft intends for end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner that directly infringes the '794 patent.

42. On information and belief, as of the date of the Complaint (D.I. 1) Microsoft had and has knowledge that the induced acts of end users of at least Bing Maps, the Bing Maps Preview application, and the Bing Search application constitute infringement of the '794 patent.

43. As of the date of the Complaint (D.I. 1), Microsoft is contributing to direct infringement of at least claim 2 of the ’794 patent under 35 U.S.C. § 271(c).

44. Bing Maps and Bing Maps Preview, at least, are designed to operate in a manner that is covered by one or more claims of the ’794 patent, through, among other things, their use of map tiles. *See, e.g.*, Joe Schwartz , *Bing Maps Tile System*, Microsoft Developer Network, http://msdn.microsoft.com/en-us/library/bb259689.aspx (last visited Oct. 31, 2014), which includes a graphic that describes Bing Maps map tiles. (*See* Paragraph 34.)

45. Microsoft mapping products and services, including at least Bing Maps and Bing Maps Preview, are not staple articles of commerce with substantial non-infringing uses.

46. When an end user uses Microsoft mapping products and services such as Bing Maps and Bing Maps Preview in their intended manner, such user performs all of the steps of one or more claims of the ’794 patent, including at least claim 2, as the claims are properly construed both prior to and after the issuance of the Certificate of Correction. Accordingly, at least Bing Maps and Bing Maps Preview are a material part of the invention claimed in the ’794 patent.

47. The acts of direct and indirect infringement by Microsoft have caused, are causing, and will cause damage to Bradium. Bradium is entitled to recover such damages from Microsoft, in an amount subject to proof at trial.

COUNT II
INFRINGEMENT OF U.S. PATENT NO. 7,908,343

48. The contents of Paragraphs 1 through 47 are incorporated by reference as if specifically set forth herein.

49. On March 15, 2011, United States Letters Patent No. 7,908,343 for "Optimized Image Delivery Over Limited Bandwidth Communication Channels," was duly and legally issued to Isaac Levanon and Yoni Lavi. All rights and interest in the '343 patent have been assigned to Bradium, including the right to sue for past damages. A true and correct copy of the '343 patent is attached hereto as Exhibit D.

50. On information and belief, Microsoft has directly infringed and continues to infringe at least claims 1 and 13 of the '343 patent. The infringing acts include performing each step of claim 1 by or under the direction and control of Microsoft, including the performance of each step by, at least, using one or more of: (i) Microsoft's Bing Search application for Android on a computer device such as a smartphone in combination with Microsoft servers; (ii) Microsoft's Bing Search application for iOS on a computer device such as a smartphone in combination with Microsoft servers; (iii) Microsoft's Bing Maps Preview application for Microsoft Windows on a computer device such as a smartphone in combination with Microsoft servers; and/or (iv) Microsoft's Bing Maps web sites available via smartphone web browsers on a computer device such as a smartphone in combination with Microsoft servers.

51. The infringing acts further include at least making, using, putting into service, selling, offering for sale, and/or importing, systems that are covered by one or more claims of the '343 patent, including claim 13, for example, at least (i) Microsoft's Bing Search application for Android on a computer device such as a smartphone in combination with Microsoft servers; (ii) Microsoft's Bing Search application for iOS on a computer device such as a smartphone in combination with Microsoft servers; (iii) Microsoft's Bing Maps Preview application for Windows on a computer device such as a smartphone in combination with Microsoft servers; and

(iv) Microsoft's Bing Maps web sites available via at least smartphone web browsers on a computer device such as a smartphone in combination with Microsoft servers.

52. For example, on information and belief, the Bing Search application, operating on an Android device, requests, receives, and displays Bing Maps map tiles based on a user's image viewpoint. Further, on information and belief, Microsoft servers process Bing Maps map tiles in an infringing manner.

53. Microsoft is therefore liable for direct infringement of the '343 patent pursuant to 35 U.S.C. § 271(a).

54. As of the date of the Complaint (D.I. 1), Microsoft is inducing infringement of at least claim 13 of the '343 patent under 35 U.S.C. § 271(b), because Microsoft has intended, and continues to intend, to cause end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner covered by one or more claims of the '343 patent, including claim 13.

55. Microsoft instructs and encourages end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner covered by one or more claims of the '343 patent, including claim 13.

56. On information and belief, end users have used, and continue to use, at least Bing Search, Bing Maps Preview, and Bing Maps in an infringing manner, including by making, using and putting into service the system claimed in claim 13.

57. As of the date of the Complaint (D.I. 1), Microsoft is actively inducing and encouraging direct infringement by end users by offering users Bing Rewards for using

Microsoft products and services, including the Bing Search application. Microsoft provides instructions that encourage end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner covered by one or more claims of the ’343 patent, including claim 13.

58. On information and belief, Microsoft has knowledge of the ’343 patent and knowledge that end users’ use of at least Bing Maps, Bing Maps Preview, and the Bing Search application infringes the ’343 patent. On information and belief, Microsoft intends for end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner that directly infringes the ’343 patent.

59. On information and belief, as of the date of the Complaint (D.I. 1), Microsoft had and has knowledge that the induced acts of end users of at least Bing Maps, Bing Maps Preview, and the Bing Search application constitute infringement of the ’343 patent.

60. Microsoft has, and continues to, commit acts of patent infringement of at least claim 13 of the ’343 patent under 35 U.S.C. § 271(f). Microsoft has, and continues to, supply all and/or a substantial portion of the components of the invention of at least claim 13 of the ’343 patent in or from the United States, in such a manner as to actively induce the combination of such components outside of the United States in a manner that would infringe the patent if such combination occurred within the United States.

61. Further, Microsoft has, and continues to, supply one or more components of the invention of at least claim 13 of the ’343 patent that are especially adapted for use in the invention and are not a staple article of commerce suitable for substantial noninfringing use, knowing that such components are so adapted and intending that such components will be

combined outside of the United States in a manner that would infringe at least claim 13 of the ’343 patent if such combination occurred within the United States.

62. For example, on information and belief, Microsoft supplies Microsoft hardware and/or servers in or from the United States, including as part of the Microsoft Edge Caching Program, that are especially adapted to, for example, deliver Bing Maps tiles to at least Microsoft’s Bing Search application when the application is resident on a computer device such as a smartphone. When configured to provide mapping products or services, such as by providing Bing Maps tiles, the Microsoft hardware is not a staple article of commerce suitable for substantial noninfringing use. Microsoft actively induces end users to use, for example, at least the Bing Maps Preview application such that the application is combined with the Microsoft hardware and/or servers outside of the United States in a manner that would infringe at least claim 13 of the ’343 patent if such combination occurred within the United States.

63. As another example, Microsoft supplies in or from the United States, via the Windows Store, at least the Bing Maps Preview application, which is especially adapted to, among other things, retrieve Bing Maps tiles. Microsoft supplies the Bing Maps Preview application knowing that it will be combined outside of the United States with, for example, Microsoft servers, in a manner that would infringe at least claim 13 of the ’343 patent if such combination occurred within the United States.

64. As of the date of the Complaint (D.I. 1), Microsoft is contributing to direct infringement of at least claim 13 of the ’343 patent under 35 U.S.C. § 271(c).

65. Bing Maps and Bing Maps Preview, at least, are designed to operate in a manner that is covered by one or more claims of the ’343 patent, through, among other things, their use of Bing Maps map tiles. *See*, Paragraph 34, above.

66. Microsoft mapping products and services, such as Bing Maps and Bing Maps Preview, are not staple articles of commerce with substantial non-infringing uses.

67. When an end user uses Microsoft mapping products and services, such as Bing Maps and Bing Maps Preview, in their intended manner, such user makes, uses and/or puts into service the system of one or more claims of the ’343 patent, including claim 13. Accordingly, at least Bing Maps and Bing Maps Preview are a material part of the invention claimed in the ’343 patent.

68. The acts of direct and indirect infringement by Microsoft have caused, are causing, and will cause damage to Bradium. Bradium is entitled to recover such damages from Microsoft, in an amount subject to proof at trial.

COUNT III
INFRINGEMENT OF U.S. PATENT NO. 8,924,506

69. The contents of Paragraphs 1 through 68 are incorporated by reference as if specifically set forth herein.

70. On December 30, 2014, United States Letters Patent No. 8,924,506 for “Optimized Image Delivery Over Limited Bandwidth Communication Channels,” was duly and legally issued to Isaac Levanon and Yoni Lavi. All rights and interest in the ’506 patent have been assigned to Bradium, including the right to sue for past damages. A true and correct copy of the ’506 patent is attached hereto as Exhibit E.

71. On July 21, 2011, United States Patent Application Publication No. 2011/0175914 was published. A true and correct copy of Publication No. 2011/0175914 is attached hereto as Exhibit F.

72. Microsoft had prior actual notice of Application No. 13/027,929, which is the patent application that issued as the ’506 patent, and of Publication No. US 2011/0175914, which is the publication associated with this application. The claims of Publication No. US 2011/0175914 are substantially identical to the claims of the ’506 patent.

73. On information and belief, Microsoft has directly infringed and continues to infringe at least claims 1, 8 and 15 of the ’506 patent. The infringing acts include performing each step of claim 1 by or under the direction and control of Microsoft, including the performance of each step by, at least, using one or more of: (i) Microsoft’s Bing Search application for Android on a computer device such as a smartphone in combination with Microsoft servers; (ii) Microsoft’s Bing Search application for iOS on a computer device such as a smartphone in combination with Microsoft servers; (iii) Microsoft’s Bing Maps Preview application for Microsoft Windows on a computer device such as a smartphone in combination with Microsoft servers; and/or (iv) Microsoft’s Bing Maps web sites available via smartphone web browsers on a computer device such as a smartphone in combination with Microsoft servers.

74. The infringing acts further include at least making, using, putting into service, selling, offering for sale, and/or importing, systems that are covered by one or more claims of the ’506 patent, including claim 8, for example, at least (i) Microsoft’s Bing Search application for Android on a computer device such as a smartphone in combination with Microsoft servers; (ii) Microsoft’s Bing Search application for iOS on a computer device such as a smartphone in

combination with Microsoft servers; (iii) Microsoft's Bing Maps Preview application for Windows on a computer device such as a smartphone in combination with Microsoft servers; and/or (iv) Microsoft's Bing Maps web sites available via smartphone web browsers on a computer device such as a smartphone in combination with Microsoft servers.

75. The infringing acts further include at least making, using, putting into service, selling, offering for sale, and/or importing, computers that are covered by one or more claims of the '506 patent, including claim 15, for example, at least Microsoft servers that work with Bing Search applications, Bing Maps Preview applications, and/or Bing Maps web sites.

76. Microsoft is therefore liable for direct infringement of the '506 patent pursuant to 35 U.S.C. § 271(a).

77. As of the date of the Complaint (D.I. 1), Microsoft is inducing infringement of at least claims 8 and 15 of the '506 patent under 35 U.S.C. § 271(b), because Microsoft has intended, and continues to intend, to cause end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner covered by one or more claims of the '506 patent, including claims 8 and 15.

78. Microsoft instructs and encourages end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner covered by one or more claims of the '506 patent, including claims 8 and 15.

79. On information and belief, end users have used, and continue to use, at least Bing Search, Bing Maps Preview, and Bing Maps in an infringing manner, including by making, using and putting into service the system claimed in claim 8 and the computer claimed in claim 15.

80. As of the date of the Complaint (D.I. 1), Microsoft is actively inducing and encouraging direct infringement by end users by offering users Bing Rewards for using Microsoft products and services, including the Bing Search application. Microsoft provides instructions that encourage end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner covered by one or more claims of the ’506 patent, including claims 8 and 15.

81. On information and belief, Microsoft has knowledge of the ’506 patent and knowledge that end users’ use of at least Bing Maps, Bing Maps Preview, and the Bing Search application infringes the ’506 patent. On information and belief, Microsoft intends for end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner that directly infringes the ’506 patent.

82. On information and belief, as of the date of the Complaint (D.I. 1), Microsoft had and has knowledge that the induced acts of end users of at least Bing Maps, Bing Maps Preview, and the Bing Search application constitute infringement of the ’506 patent.

83. Microsoft has, and continues to, commit acts of patent infringement of at least claim 8 of the ’506 patent under 35 U.S.C. § 271(f). Microsoft has, and continues to, supply all and/or a substantial portion of the components of the invention of at least claim 8 of the ’506 patent in or from the United States, in such a manner as to actively induce the combination of such components outside of the United States in a manner that would infringe the patent if such combination occurred within the United States.

84. Further, Microsoft has, and continues to, supply one or more components of the invention of at least claim 8 of the ’506 patent that are especially adapted for use in the invention

and are not a staple article of commerce suitable for substantial noninfringing use, knowing that such components are so adapted and intending that such components will be combined outside of the United States in a manner that would infringe at least claim 8 of the ’506 patent if such combination occurred within the United States.

85. For example, on information and belief, Microsoft supplies Microsoft hardware and/or servers in or from the United States, including as part of the Microsoft Edge Caching Program, that are especially adapted to, for example, deliver Bing Maps tiles to at least Microsoft's Bing Search application when the application is resident on a computer device such as a smartphone. When configured to provide mapping products or services, such as by providing Bing Maps tiles, the Microsoft hardware is not a staple article of commerce suitable for substantial noninfringing use. Microsoft actively induces end users to use, for example, at least the Bing Maps Preview application such that the application is combined with the Microsoft hardware and/or servers outside of the United States in a manner that would infringe at least claim 8 of the ’506 patent if such combination occurred within the United States.

86. As another example, Microsoft supplies in or from the United States, via the Windows Store, at least the Bing Maps Preview application, which is especially adapted to, among other things, retrieve Bing Maps tiles. Microsoft supplies the Bing Maps Preview application knowing that it will be combined outside of the United States with, for example, Microsoft servers, in a manner that would infringe at least claim 8 of the ’506 patent if such combination occurred within the United States.

87. As of the date of the Complaint (D.I. 1), Microsoft is contributing to direct infringement of at least claims 8 and 15 of the ’506 patent under 35 U.S.C. § 271(c).

88. Bing Maps and Bing Maps Preview, at least, are designed to operate in a manner that is covered by one or more claims of the ’506 patent, through, among other things, their use of Bing Maps map tiles. See, Paragraph 34, above.

89. Microsoft mapping products and services, such as Bing Maps and Bing Maps Preview, are not staple articles of commerce with substantial non-infringing uses.

90. When an end user uses Microsoft mapping products and services, such as Bing Maps and Bing Maps Preview, in their intended manner, such user makes, uses and/or puts into service the system of one or more claims of the ’506 patent, including claim 8, and the computer of one or more claims of the ’506 patent, including claim 15. Accordingly, at least Bing Maps and Bing Maps Preview are a material part of the invention claimed in the ’506 patent.

91. Microsoft is liable for infringement of Bradium’s provisional rights in at least claims 1, 8 and 15 of the ’506 patent for the same reasons that Microsoft is liable for infringement of at least claims 1, 8, and 15 of the ’506 patent.

92. The acts of direct and indirect infringement, infringement of provisional rights, and the supply of components in or from the United States by Microsoft have caused, are causing, and will cause damage to Bradium. Bradium is entitled to recover such damages from Microsoft, in an amount subject to proof at trial.

COUNT IV
INFRINGEMENT OF U.S. PATENT NO. 9,253,239

93. The contents of Paragraphs 1 through 92 are incorporated by reference as if specifically set forth herein.

94. On February 2, 2016, United States Letters Patent No. 9,253,239 for "Optimized Image Delivery Over Limited Bandwidth Communication Channels," was duly and legally issued to Isaac Levanon and Yoni Lavi. All rights and interest in the '239 patent have been assigned to Bradium, including the right to sue for past damages. A true and correct copy of the '239 patent is attached hereto as Exhibit G.

95. Microsoft has directly infringed and continues to directly infringe at least claim 1 of the '239 patent. The infringing acts include performing each step of claim 1 by or under the direction and control of Microsoft, including the performance of each step by, at least, using one or more of: (i) Microsoft's Bing Search application for Android on a computer device such as a smartphone in combination with Microsoft servers; (ii) Microsoft's Bing Search application for iOS on a computer device such as a smartphone in combination with Microsoft servers; (iii) Microsoft's Bing Maps Preview application for Microsoft Windows on a computer device such as a smartphone in combination with Microsoft servers; and/or (iv) Microsoft's Bing Maps web sites available via smartphone web browsers on a computer device such as a smartphone in combination with Microsoft servers.

96. As an example, claim 1 of the '239 patent recites "issuing a first request from the user computing device to one or more servers, over one or more network communication channels, the first request being for a first update data parcel corresponding to a first derivative image of a predetermined image, the predetermined image corresponding to source image data."

97. Microsoft provides Bing Maps through, for example, a web browser or the Bing Search application for use on a user computing device such as an iOS device or Android device connected to a network communication channel such as, for example, the Internet. For example,

as the user browses to a new area of a virtual map, the browser issues requests for a new map tile to, for example, content delivery network servers controlled by Microsoft. *See, e.g.*, Bing Maps Tile System, MICROSOFT, https://msdn.microsoft.com/en-us/library/bb259689.aspx (last visited January 27, 2016).

98. As another example, claim 1 of the ’239 patent recites “the first update data parcel is selected based on a first user-controlled image viewpoint on the user computing device relative to the predetermined image.”

99. As the user using Bing Maps pans and zooms to a new area of a virtual map, the browser issues requests for a new map tile based on the user-controlled image viewpoint.

100. As a further example, claim 1 of the ’239 patent recites “receiving the first update data parcel at the user computing device from the one or more servers over the one or more network communication channels.”

101. For example, after a request is issued from the user computing device, the user computing device receives a map tile from the one or more, for example, content delivery network servers controlled by Microsoft over the network communication channel.

102. As a further example, claim 1 of the ’239 patent recites “displaying the first discrete portion on the user computing device using the first update data parcel.”

103. For example, the user computing device displays the map tile on the device.

104. Bing Maps, for example as provided via a web browser or the Bing Search application for use on a user computing device, makes multiple requests for map tiles and, as a

result, receives and displays multiple map tiles in response to, for example, changes in user-controlled image viewpoint.

105. Microsoft is therefore liable for direct infringement of the ’239 patent pursuant to 35 U.S.C. § 271(a).

106. Microsoft has, and continues to, induce infringement of at least claim 1 of the ’239 patent under 35 U.S.C. § 271(b) because Microsoft has intended, and continues to intend, to cause end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner covered by one or more claims of the ’239 patent, including claim 1.

107. Microsoft instructs and encourages end users to use at least Bing Maps and the Bing Search application in a manner covered by one or more claims of the ’239 patent, including claim 1. Among other things, Bing Maps includes tools, icons and other insignia to invite users to use the underlying functions that allow the user to control the image viewpoint and to obtain map images in an infringing manner.

108. On information and belief, end users have used, and continue to use, at least, Bing Search, Bing Maps Preview, and Bing Maps in an infringing manner.

109. On information and belief, Microsoft has been and is actively inducing and encouraging direct infringement by end users by offering users Bing Rewards for using Microsoft and/or Bing products and services, including the Bing Search application. Microsoft also provides instructions to encourage end users to use at least Bing Maps, Bing Maps Preview and the Bing Search application in a manner covered by one or more claims of the ’239 patent, including claim 1.

110. On information and belief, Microsoft had (both prior to and after February 2, 2016) and has knowledge of the ’239 patent and knowledge that end users’ use of at least Bing Maps, Bing Maps Preview, and the Bing Search application infringes the ’239 patent. On information and belief, Microsoft intends for end users to use at least Bing Maps, Bing Maps Preview, and the Bing Search application in a manner that directly infringes the ’239 patent.

111. On information and belief, Microsoft had and has knowledge that the induced acts of end users of at least Bing Maps, the Bing Maps Preview application, and the Bing Search application constitute infringement of the ’239 patent.

112. Microsoft has contributed to, and continues to contribute to, direct infringement of at least claim 1 of the ’239 patent under 35 U.S.C. § 271(c).

113. Bing Maps and Bing Maps Preview, at least, are designed to operate in a manner that is covered by one or more claims of the ’239 patent, through, among other things, their use of map tiles. *See, e.g.*, Joe Schwartz , *Bing Maps Tile System*, Microsoft Developer Network, http://msdn.microsoft.com/en-us/library/bb259689.aspx (last visited February 19, 2016), which includes a graphic that describes Bing Maps map tiles. (*See* Paragraph 34.)

114. Microsoft mapping products and services, including at least Bing Maps and Bing Maps Preview, are not staple articles of commerce with substantial non-infringing uses. Bing Maps, for example, is designed to operate in conjunction with map tiles that are requested and provided from a server.

115. When an end user uses Microsoft mapping products and services such as Bing Maps and Bing Maps Preview in their intended manner, such user performs all of the steps of

one or more claims of the ’239 patent, including at least claim 1. Bing Maps, for example, operates by requesting and receiving multiple map tiles from a server. Accordingly, at least Bing Maps and Bing Maps Preview are a material part of the invention claimed in the ’239 patent.

116. Microsoft has infringed and continues to infringe despite an objectively high likelihood that its actions constitute infringement of Bradium’s valid patent rights. On information and belief, Microsoft knew of or should have known of this objectively high risk at least as early as it commenced its infringing activities and/or it became aware of the ’239 patent, as described above. Bradium, for example, informed Microsoft that it was and is infringing. Thus, Microsoft’s infringement of the ’239 patent has been and continues to be willful.

117. The acts of direct and indirect infringement by Microsoft have caused, are causing, and will cause damage to Bradium. Bradium is entitled to recover such damages from Microsoft, in an amount subject to proof at trial.

PRAYER FOR RELIEF

WHEREFORE, Bradium prays for judgment and seeks relief as follows:

(a) For judgment that Microsoft has infringed one or more claims of the ’794 patent, both prior to and after the issuance of the Certificate of Correction, and continues to infringe;

(b) For judgment that Microsoft has infringed, and continues to infringe, one or more of claims of the ’343 patent;

(c) For judgment that Microsoft has infringed, and continues to infringe, one or more of claims of the ’506 patent and one or more claims of Publication No. 2011/0175914;

(d) For judgment that Microsoft has infringed, and continues to infringe, one or more of claims of the ’239 patent;

(e) For damages, together with prejudgment and post-judgment interest, and post-judgment royalties, sustained by Bradium as the result of Microsoft’s acts of infringement, of no less than a reasonable royalty, and for treble damages based on Microsoft’s willful infringement of the ’239 patent;

(f) For pre-issuance royalties for the ’506 patent under 35 U.S.C. § 154(d)(1);

(g) For a permanent injunction enjoining Microsoft and its officers, directors, agents, servants, affiliates, employees, divisions, branches, subsidiaries, parents, and all others acting in active concert or participation with them, from infringing the claims of the ’794 patent, the ’343 patent, the ’506 patent, and the ’239 patent;

(h) For judgment that this case is exceptional and for an award of attorney’s fees, costs, and expenses pursuant to 35 U.S.C. § 285 or as otherwise permitted by law;

(i) For all costs of suit; and

(j) For such other and further relief as the Court may deem just and proper.

JURY DEMAND

Pursuant to Fed. R. Civ. P. 38, Plaintiff demands a trial by jury of any and all issues properly triable by a jury.

Dated: March 11, 2016

Respectfully Submitted:

PHILLIPS GOLDMAN MCLAUGHLIN & HALL, P.A.

By: */s/ John C. Phillips, Jr.*
John C. Phillips, Jr. (#110)
David A. Bilson (#4986)
1200 North Broom Street
Wilmington, DE 19806
(302) 655-4200
jcp@pgmhlaw.com
dab@pgmhlaw.com

Attorneys for Plaintiff, Bradium Technologies LLC

Of Counsel:

Michael N. Zachary
KENYON & KENYON LLP
1801 Page Mill Road, Suite 210
Palo Alto, CA 94304
Telephone: 1.650.384.4700
Facsimile: 1.650.384.4701
Email: mzachary@kenyon.com

Christopher J. Coulson
KENYON & KENYON LLP
One Broadway
New York, NY 10004
Telephone: 212.425.7200
Facsimile: 212.425.5288
Email: ccoulson@kenyon.com

EXHIBIT A

To the First Amended Complaint

GENERAL PATENT
CORPORATION

May 27, 2014

Brad Smith, Esq.
Executive Vice President and General Counsel
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-639

Via Priority Mail (with enclosures) and
Facsimile: (425) 708-0096 (letter only)

Re: Bradium Technologies LLC Patent Portfolio

Dear Mr. Smith:

We are the licensing agent for Bradium Technologies LLC (Bradium).

I write on behalf of Bradium to make you aware of a portfolio of three U.S. Patents and one U.S. Patent Application, which are listed below, that is available for licensing.

- U.S. Patent No. **7,139,794**, Levanon et al., "System and Methods for Network Image Delivery with Dynamic Viewing Frustum Optimized for Limited Bandwidth Communication Channels";
- U.S. Patent No. **7,644,131**, Levanon et al., "Optimized Image Delivery Over Limited Bandwidth Communication Channels";
- U.S. Patent No. **7,908,343**, Levanon et al., "Optimized Image Delivery Over Limited Bandwidth Communication Channels";
- U.S. Patent Application No. **13/027,929**, Levanon et al., "Optimized Image Delivery Over Limited Bandwidth Communication Channels".

For your convenience and reference, I enclose copies of each of the three U.S. Patents.

If Microsoft Corporation has any interest in starting discussions regarding licensing or learning more about the portfolio, please let me know at your

Montebello Park
75 Montebello Road
Suffern, NY 10901-3746
tel 845.368.4000
fax 845.368.8770
www.generalpatent.com

convenience. I can be reached at (845) 368-4000 ext. 107 or at kingham@generalpatent.com.

Sincerely,

GENERAL PATENT CORPORATION

Kathlene Ingham

Kathlene Ingham
Director of Licensing

:kpi
Enclosures

GENERAL PATENT
CORPORATION

December 23, 2014

Brad Smith, Esq.
Executive Vice President and General Counsel
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-639

Via Priority Mail (with enclosures) and
Facsimile: (425) 708-0096 (letter only)

Re: Bradium Technologies LLC Patent Portfolio

Dear Mr. Smith:

I write as a follow up to my letter dated May 27, 2014 regarding a licensing proposal for the Bradium Technologies LLC patent portfolio; a copy of that letter is enclosed for your ease of reference. To date, we have not received a substantive response to our letter.

There have been several developments with respect to the Bradium portfolio since our last letter. The United States Patent and Trademark Office issued a Certificate of Correction for U.S. Patent No. 7,139,794 (the "'794 patent") on July 1, 2014. For your convenience and reference, I enclose a copy of the '794 patent that includes the Certificate of Correction. Also, we have received notice that Application No. 13/027,929, which I referenced in my May 27, 2014 letter, will issue on December 30, 2014 as U.S. Patent No. 8,924,506. I also attach a copy of this issue notification for your convenience and reference, as well as the publication that is associated with this application, Publication No. US 2011/0175914.

If Microsoft Corporation has any interest in starting discussions regarding licensing or learning more about the portfolio, please let me know at your convenience, as we remain interested in licensing discussions. I can be reached at (854) 368-4000 ext. 107 or at kingham@generalpatent.com.

Sincerely,

GENERAL PATENT CORPORATION

Kathlene Ingham

Kathlene Ingham
Director of Licensing

:kpi
Enclosures

PATENT LICENSING

PATENT ENFORCEMENT

INTELLECTUAL PROPERTY STRATEGY

INTELLECTUAL PROPERTY MANAGEMENT

INTELLECTUAL PROPERTY VALUATION

Montebello Park
75 Montebello Road
Suffern, NY 10901-3746

tel 845.368.4000
fax 845.368.8770
www.generalpatent.com

EXHIBIT B

To the First Amended Complaint

Bradium Technologies

February 2, 2016

Brad Smith, Esq.
Chief Legal Officer
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

Via Federal Express (with enclosure)
And Facsimile: (425) 708-0096 (letter only)

Re: Bradium Technologies LLC Patent Portfolio

Dear Mr. Smith:

I write to provide notice of infringement of a patent owned by Bradium Technologies LLC ("Bradium") that issued today.

In particular, please be advised that the United States Patent and Trademark Office today issued U.S Patent No. 9,253,239 (the "'239 Patent"). Bradium is the assignee of all rights regarding the '239 Patent. For your convenience and reference, a copy of the '239 Patent is included with this letter.

Bradium believes that Microsoft is directly infringing the '239 Patent at least by putting into service and/or using Bing Search, Bing Maps Preview and Bing Maps products and/or services in the United States.

Bradium also believes that use of Bing Search, Bing Maps Preview and Bing Maps products and/or services in the United States directly infringes the '239 Patent. Bradium therefore believes that Microsoft, at least by supplying Bing Search and Bing Maps products and/or services and operating Bing Search and Bing Maps servers, is contributing to infringement of the '239 Patent by users, such as end users, in the United States.

Furthermore, Bradium believes that Microsoft, by among other things providing instructions that encourage use of Bing Maps, Bing Maps Preview and Bing Search products and/or services, is inducing infringement of the '239 Patent by users, including end users, in the United States.

Bradium requests that Microsoft immediately cease its infringing activities regarding the '239 Patent. Bradium further requests that Microsoft provide a response by no later than February 16, 2016, to indicate whether it will comply with Bradium's request to cease Microsoft's infringing activities.

Sincerely,

BRADIUM TECHNOLOGIES LLC

Michael E. Shanahan, Esq.
General Counsel

Enclosures (1)

CC: M. Zachary, Esq.
C. Coulson, Esq.

EXHIBIT C

To the First Amended Complaint

US007139794B2

(12) **United States Patent**
Levanon et al.

(10) **Patent No.: US 7,139,794 B2**
(45) **Date of Patent: Nov. 21, 2006**

(54) **SYSTEM AND METHODS FOR NETWORK IMAGE DELIVERY WITH DYNAMIC VIEWING FRUSTUM OPTIMIZED FOR LIMITED BANDWIDTH COMMUNICATION CHANNELS**

(75) Inventors: **Isaac Levanon**, Ramat Hasharn (IL); **Yoni Lavi**, Raanana (IL)

(73) Assignee: **3-D-V-U Israel (2000) Ltd.**, Rannana (IL)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 918 days.

(21) Appl. No.: **10/035,981**

(22) Filed: **Dec. 24, 2001**

(65) **Prior Publication Data**

US 2002/0118224 A1 Aug. 29, 2002

Related U.S. Application Data

(60) Provisional application No. 60/258,488, filed on Dec. 27, 2000, provisional application No. 60/258,489, filed on Dec. 27, 2000, provisional application No. 60/258,465, filed on Dec. 27, 2000, provisional application No. 60/258,468, filed on Dec. 27, 2000, provisional application No. 60/258,466, filed on Dec. 27, 2000, provisional application No. 60/258,467, filed on Dec. 27, 2000.

(51) **Int. Cl.**
G06F 15/16 (2006.01)

(52) **U.S. Cl.** **709/203**; 709/246; 709/217; 709/219; 345/420; 345/581; 382/232; 382/235

(58) **Field of Classification Search** 709/203, 709/246–247, 217–219; 345/418–420, 581, 345/619; 382/103, 232, 235, 240
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,698,689	A *	10/1987	Tzou	358/426.14
5,966,672	A *	10/1999	Knupp	702/16
6,182,114	B1 *	1/2001	Yap et al.	709/203
6,397,259	B1 *	5/2002	Lincke et al.	709/236
6,671,424	B1 *	12/2003	Skoll et al.	382/305
6,704,024	B1 *	3/2004	Robotham et al.	345/581
6,874,012	B1 *	3/2005	St. Pierre	709/207

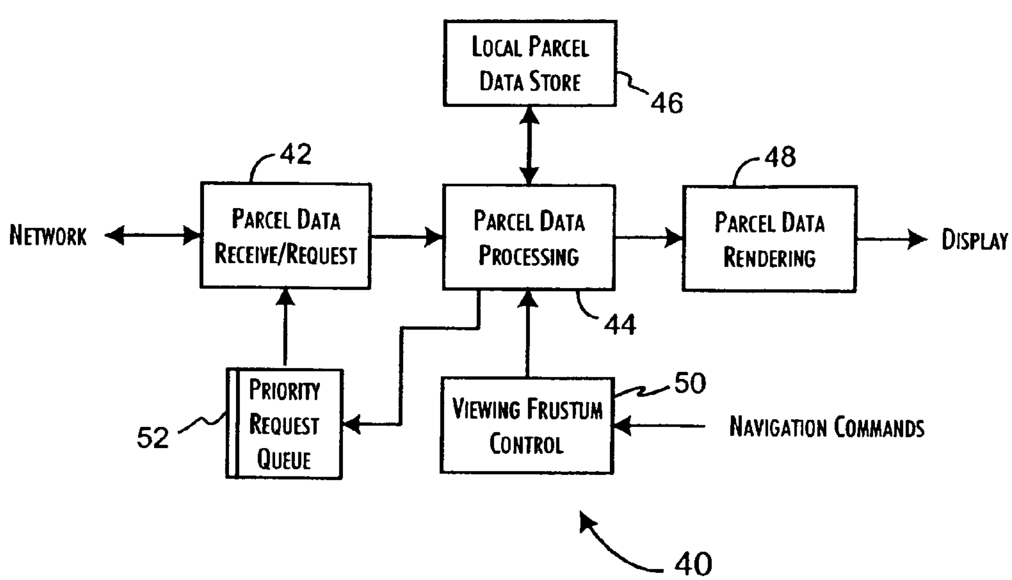
* cited by examiner

Primary Examiner—Philip B. Tran

(57) **ABSTRACT**

Dynamic visualization of image data provided through a network communications channel is performed by a client system including a parcel request subsystem and a parcel rendering subsystem. The parcel request subsystem includes a parcel request queue and is operative to request discrete image data parcels in a priority order and to store received image data parcels in a parcel data store. The parcel request subsystem is responsive to an image parcel request of assigned priority to place the image parcel request in the parcel request queue ordered in correspondence with the assigned priority. The parcel rendering subsystem is coupled to the parcel data store to selectively retrieve and render received image data parcels to a display memory. The parcel rendering system provides the parcel request subsystem with the image parcel request of the assigned priority.

2 Claims, 5 Drawing Sheets



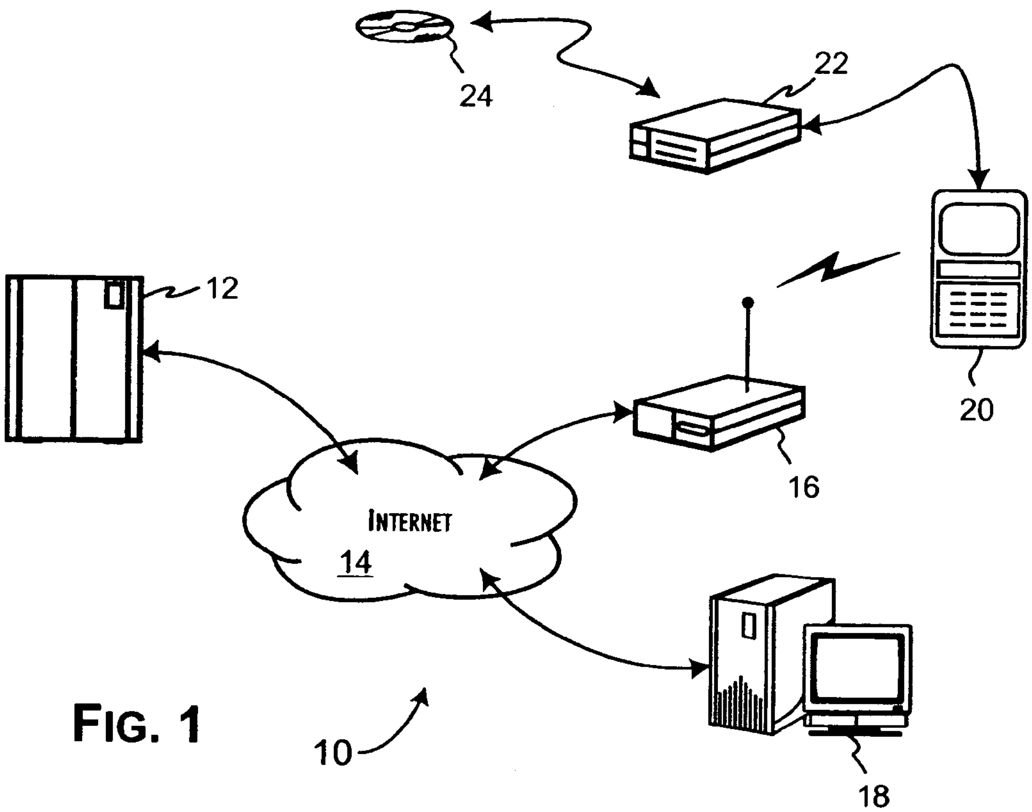

24
22
12
20
16
INTERNET
14
18
10

FIG. 1

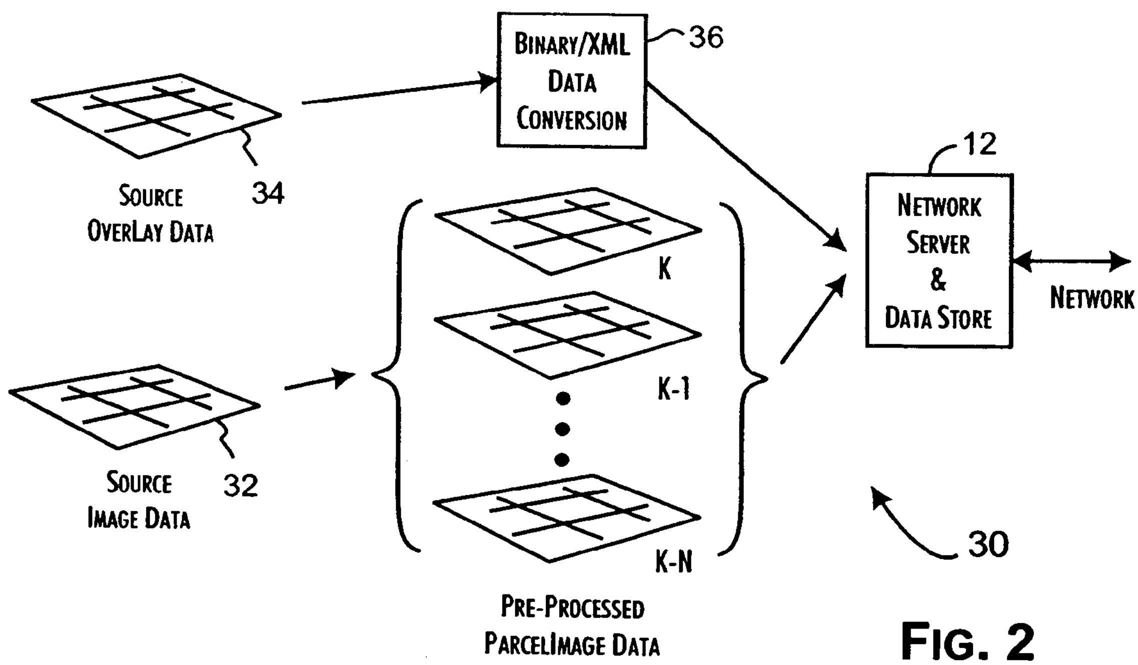
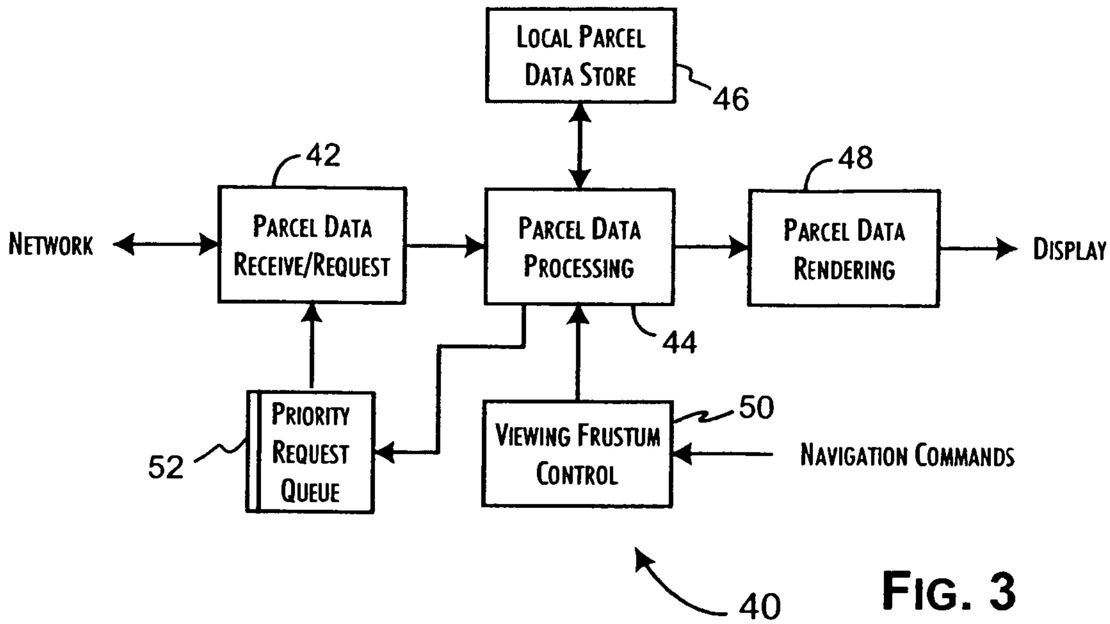
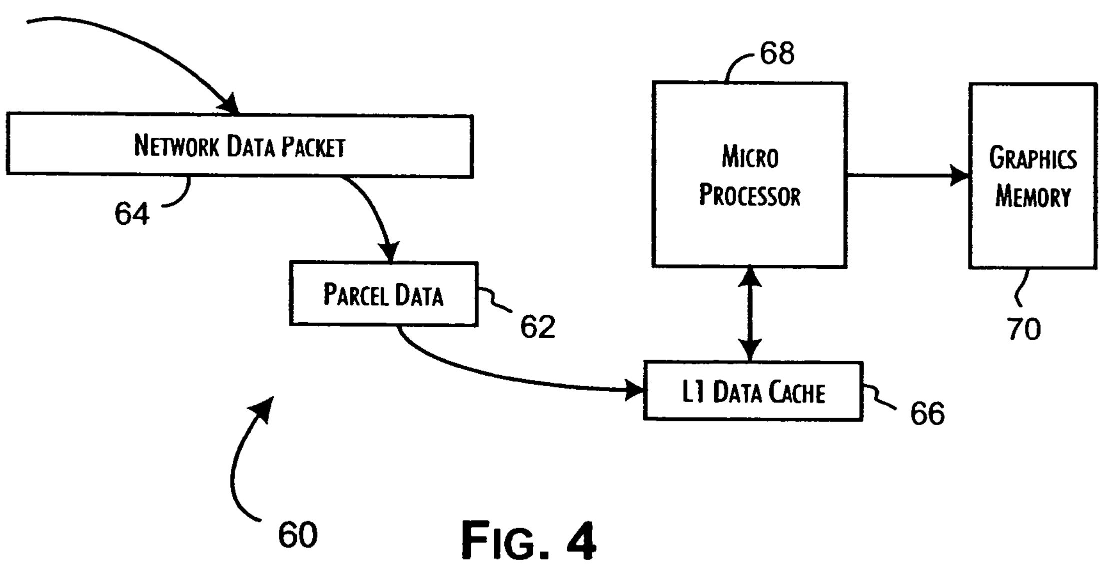

BINARY/XML
DATA
CONVERSION
36
12
SOURCE
OVERLAY DATA
34
NETWORK
SERVER
&
DATA STORE
NETWORK
K
K-1
K-N
SOURCE
IMAGE DATA
32
30
PRE-PROCESSED
PARCELIMAGE DATA

FIG. 2



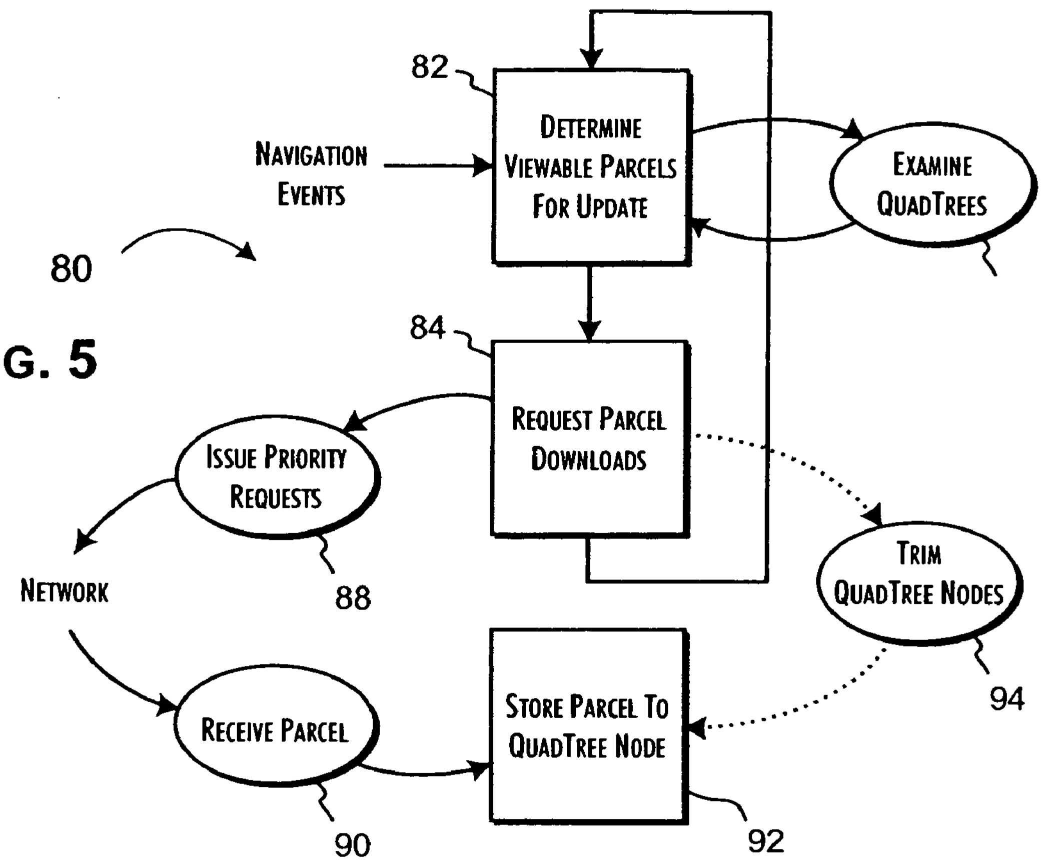
Local Parcel Data Store
46
42
48
Network
Parcel Data Receive/Request
Parcel Data Processing
Parcel Data Rendering
Display
44
50
Priority Request Queue
52
Viewing Frustum Control
Navigation Commands
40

FIG. 3



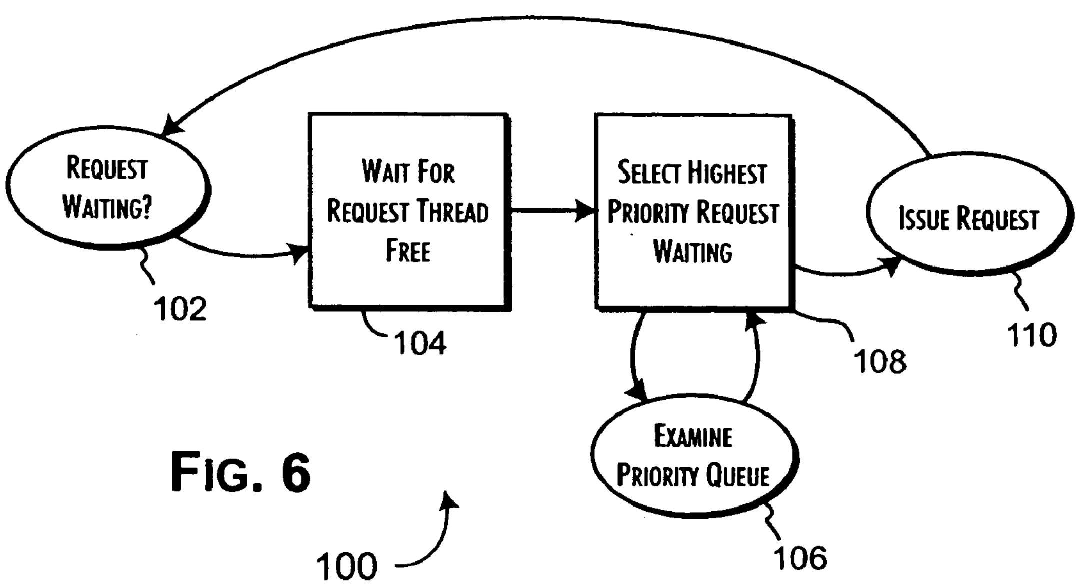
68
Network Data Packet
Micro Processor
Graphics Memory
64
Parcel Data
62
70
L1 Data Cache
66
60

FIG. 4



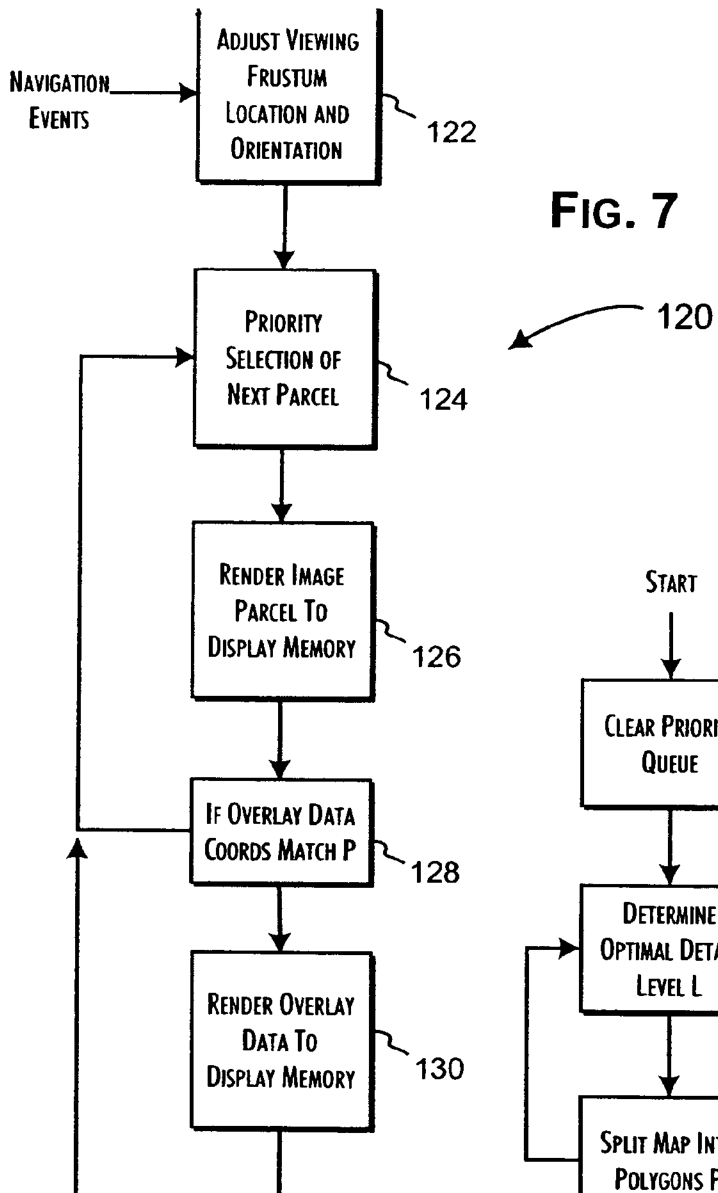
82
Navigation Events
Determine Viewable Parcels For Update
Examine QuadTrees
80
84
Request Parcel Downloads
Issue Priority Requests
Network
88
Trim QuadTree Nodes
94
Receive Parcel
Store Parcel To QuadTree Node
90
92

FIG. 5



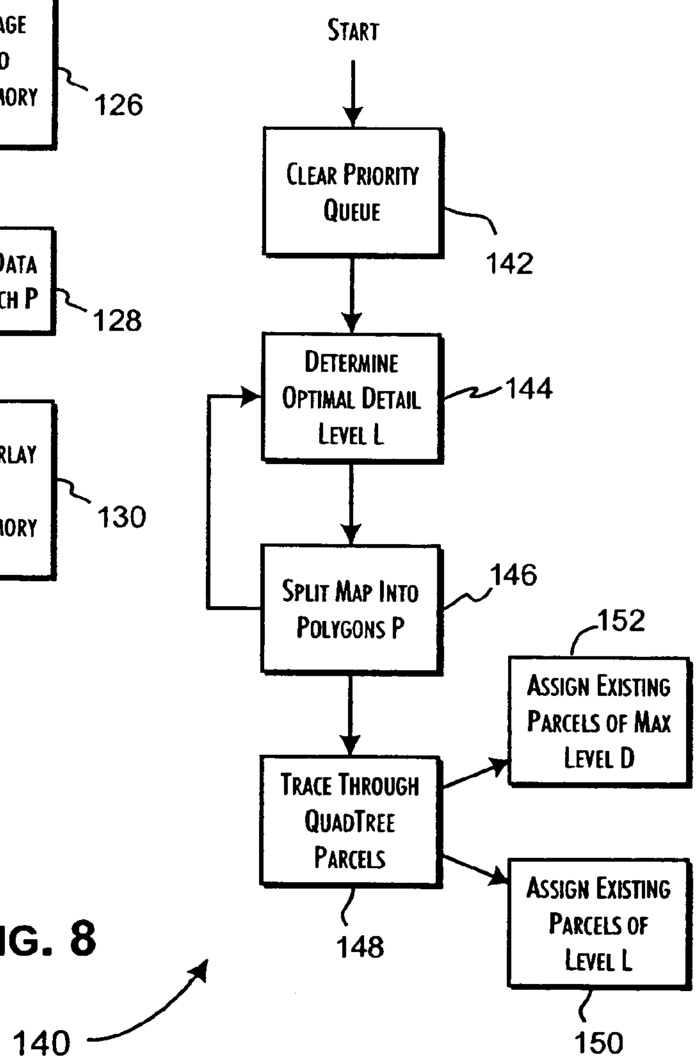
Request Waiting?
Wait For Request Thread Free
Select Highest Priority Request Waiting
Issue Request
102
104
108
110
Examine Priority Queue
106
100

FIG. 6



NAVIGATION EVENTS
ADJUST VIEWING FRUSTUM LOCATION AND ORIENTATION
122
PRIORITY SELECTION OF NEXT PARCEL
124
RENDER IMAGE PARCEL TO DISPLAY MEMORY
126
IF OVERLAY DATA COORDS MATCH P
128
RENDER OVERLAY DATA TO DISPLAY MEMORY
130
120

FIG. 7



START
CLEAR PRIORITY QUEUE
142
DETERMINE OPTIMAL DETAIL LEVEL L
144
SPLIT MAP INTO POLYGONS P
146
TRACE THROUGH QUADTREE PARCELS
148
ASSIGN EXISTING PARCELS OF MAX LEVEL D
152
ASSIGN EXISTING PARCELS OF LEVEL L
150
140

FIG. 8

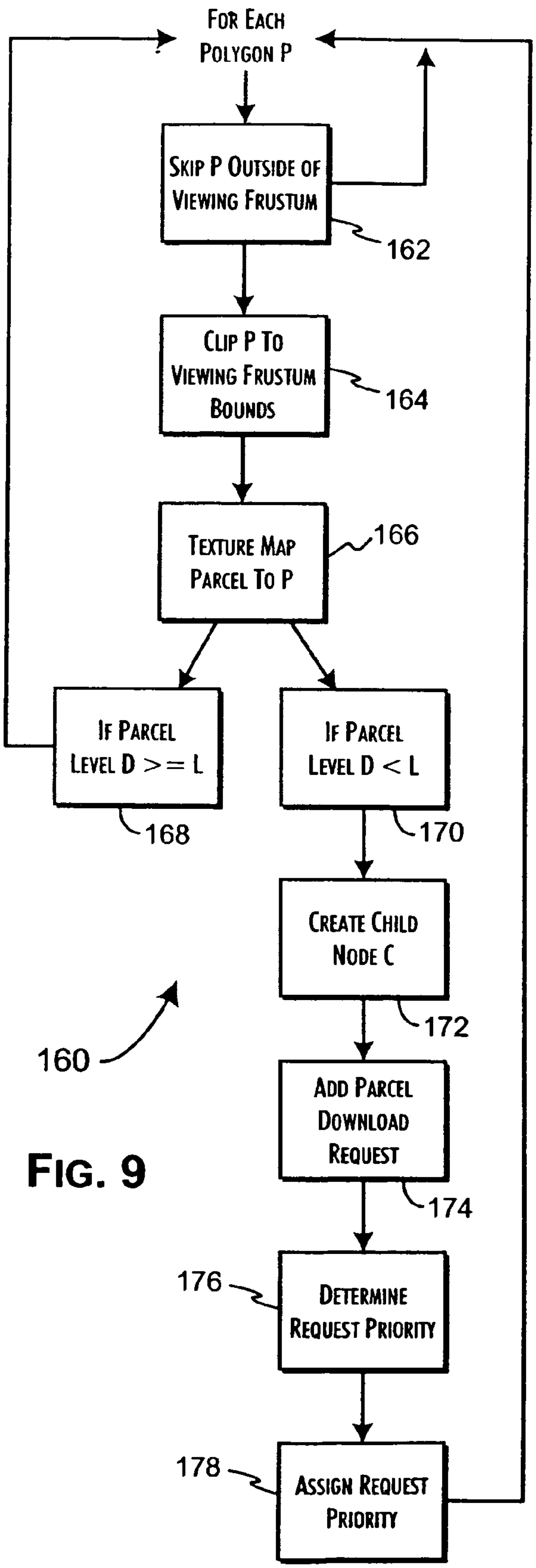

FOR EACH POLYGON P
SKIP P OUTSIDE OF VIEWING FRUSTUM
162
CLIP P TO VIEWING FRUSTUM BOUNDS
164
TEXTURE MAP PARCEL TO P
166
IF PARCEL LEVEL D >= L
168
IF PARCEL LEVEL D < L
170
CREATE CHILD NODE C
172
ADD PARCEL DOWNLOAD REQUEST
174
176
DETERMINE REQUEST PRIORITY
178
ASSIGN REQUEST PRIORITY
160

FIG. 9

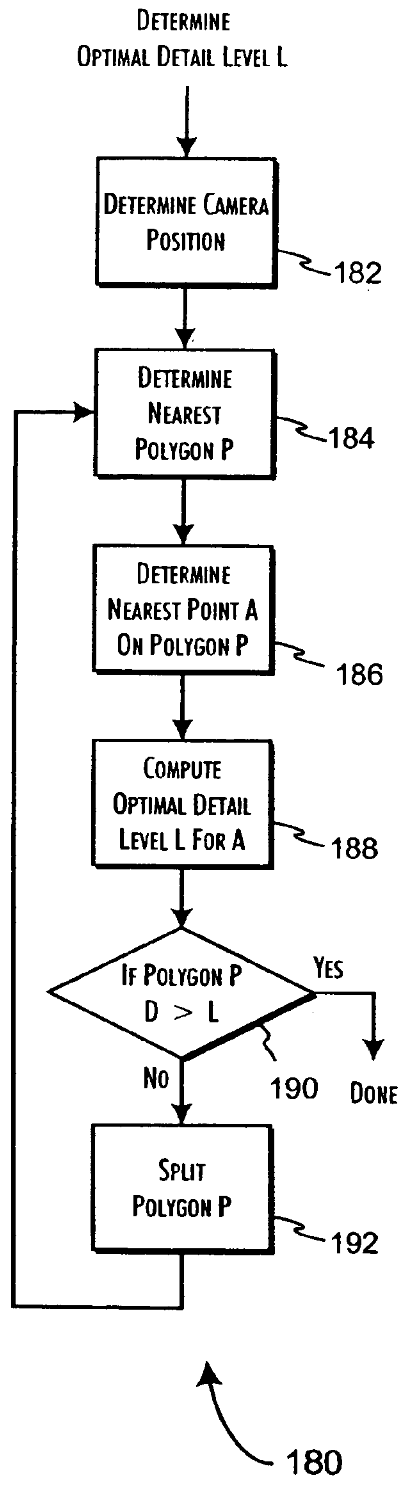

DETERMINE OPTIMAL DETAIL LEVEL L
DETERMINE CAMERA POSITION
182
DETERMINE NEAREST POLYGON P
184
DETERMINE NEAREST POINT A ON POLYGON P
186
COMPUTE OPTIMAL DETAIL LEVEL L FOR A
188
IF POLYGON P D > L
YES
NO
190
DONE
SPLIT POLYGON P
192
180

FIG. 10

SYSTEM AND METHODS FOR NETWORK IMAGE DELIVERY WITH DYNAMIC VIEWING FRUSTUM OPTIMIZED FOR LIMITED BANDWIDTH COMMUNICATION CHANNELS

This application claims the benefit of U.S. Provisional Application Nos. 60/258,488, 60/258,485, 60/258,465, 60/258,468, 60/258,466, and 60/258,467, all filed Dec. 27, 2000.

CROSS-REFERENCE TO RELATED APPLICATIONS

The present application is related to the co-pending application Optimized Image Delivery over Limited Bandwidth Communication Channels, Levanon et al., filed concurrently herewith and which is assigned to the Assignee of the present Application.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention is related to network based, image distribution systems and, in particular, to a system and methods for efficiently selecting and distributing image parcels through a narrowband or otherwise limited bandwidth communications channel to support presentation of high-resolution images subject to dynamic viewing frustums.

2. Description of the Related Art

The Internet and other network systems provide a unique opportunity to transmit complex images, typically large scale bit-maps, particularly those approaching photo-realistic levels, over large distances. In common application, the images are geographic, topographic, and other highly detailed maps. The data storage requirements and often proprietary nature of such images are such that conventional interests are to transfer the images on an as needed basis.

In conventional fixed-site applications, the image data is transferred over a relatively high-bandwidth network to client computer systems that, in turn, render the image. Client systems typically implement a local image navigation system to provide zoom and pan functions based on user interaction. As well recognized problem with such conventional systems is that full resolution image presentation is subject to the inherent transfer latency of the network. Different conventional systems have been proposed to reduce the latency affect by transmitting the image in highly compressed formats that support progressive resolution build-up of the image within the current client field of view. Using a transform compressed image transfer function increases the field of the image that can be transferred over a fixed bandwidth network in unit time. Progressive image resolution transmission, typically using a differential resolution method, permits an approximate image to be quickly presented with image details being continuously added over time.

Tzou, in U.S. Pat. No. 4,698,689, describes a two-dimensional data transform system that supports transmission of differential coefficients to represent an image. Subsequent transmitted coefficient sets are progressively accumulated with prior transmitted sets to provide a succeedingly refined image. The inverse-transform function performed by the client computer is, however, highly compute intensive. In order to simplify the transform implementation and further reduce the latency of presenting any portion of an approximate image, images are sub-divided into a regular array. This enables the inverse-transform function on the client, which is time-critical, to deal with substantially smaller coefficient data sets. The array size in Tzou is fixed, which leads to progressively larger coefficient data sets as the detail level of the image increases. Consequently, there is an inherently increasing latency in resolving finer levels of detail.

An image visualization system proposed by Yap et al., U.S. Pat. No. 6,182,114, overcomes some of the foregoing problems. The Yap et al. system also employs a progressive encoding transform to compress the image transfer stream. The transform also operates on a subdivided image, but the division is indexed to the encoding level of the transform. The encoded transform coefficient data sets are, therefore, of constant size, which supports a modest improvement in the algorithmic performance of the inverse transform operation required on the client.

Yap et al. adds utilization of client image panning or other image pointing input information to support a foveation-based operator to influence the retrieval order of the subdivided image blocks. This two-dimensional navigation information is used to identify a foveal region that is presumed to be the gaze point of a client system user. The foveation operator defines the corresponding image block as the center point of an ordered retrieval of coefficient sets representing a variable resolution image. The gaze point image block represents the area of highest image resolution, with resolution reduction as a function of distance from the gaze point determined by the foveation operator. This technique thus progressively builds image resolution at the gaze point and succeedingly outward based on a relatively compute intensive function. Shifts in the gaze point can be responded to with relative speed by preferentially retrieving coefficient sets at and near the new foveal region.

Significant problems remain in permitting the convenient and effective use of complex images by many different types of client systems, even with the improvements provided by the various conventional systems. In particular, the implementation of conventional image visualization systems is generally unworkable for smaller, often dedicated or embedded, clients where use of image visualization would clearly be beneficial. Conventional approaches effectively presume that client systems have an excess of computing performance, memory and storage. Small clients, however, typically have restricted performance processors with no dedicated floating-point support, little general purpose memory, and extremely limited persistent storage capabilities, particularly relative to common image sizes. A personal digital assistant (PDA) is a characteristic small client. Embedded, low-cost kiosk and automobile navigation systems are other typical examples. Such systems are not readily capable, if at all, of performing complex, compute-intensive Fourier or wavelet transforms, particularly within a highly restricted memory address space.

As a consequence of the presumption that the client is a substantial computing system, conventional image visualization systems also presume that the client is supported by a complete operating system. Indeed, many expect and require an extensive set of graphics abstraction layers to be provided by the client system to support the presentation of the delivered image data. In general, these abstraction layers are conventionally considered required to handle the mapping of the image data resolution to the display resolution capabilities of the client system. That is, resolution resolved image data provided to the client is unconstrained by any limitation in the client system to actually display the corresponding image. Consequently, substantial processor performance and memory can be conventionally devoted to handling image data that is not or cannot be displayed.

Another problem is that small clients are generally constrained to generally to very limited network bandwidths, particularly when operating under wireless conditions. Such limited bandwidth conditions may exist due to either the direct technological constraints dictated by the use of a low bandwidth data channel or indirect constraints imposed on relatively high-bandwidth channels by high concurrent user loads. Cellular connected PDAs and webphones are examples of small clients that are frequently constrained by limited bandwidth conditions. The conventionally realizable maximum network transmission bandwidth for such small devices may range from below one kilobit per second to several tens of kilobits per second. While Yap et al. states that the described system can work over low bandwidth lines, little more than utilizing wavelet-based data compression is advanced as permitting effective operation at low communications bandwidths. While reducing the amount of data that must be carried from the server to the client is significant, Yap et al. simply relies on the data packet transfer protocols to provide for an efficient transfer of the compressed image data. Reliable transport protocols, however, merely mask packet losses and the resultant, sometimes extended, recovery latencies. When such covered errors occur, however, the aggregate bandwidth of the connection is reduced and the client system can stall waiting for further image data to process.

Consequently, there remains a need for an image visualization system that can support small client systems, place few requirements on the supporting client hardware and software resources, and efficiently utilize low to very low bandwidth network connections.

SUMMARY OF THE INVENTION

Thus, a general purpose of the present invention is to provide an efficient system and methods of optimally presenting image data on client systems with potentially limited processing performance, resources, and communications bandwidth.

This is achieved in the present invention by providing for the dynamic visualization of image data provided through a network communications channel by a client system including a parcel request subsystem and a parcel rendering subsystem. The parcel request subsystem includes a parcel request queue and is operative to request discrete image data parcels in a priority order and to store received image data parcels in a parcel data store. The parcel request subsystem is responsive to an image parcel request of assigned priority to place the image parcel request in the parcel request queue ordered in correspondence with the assigned priority. The parcel rendering subsystem is coupled to the parcel data store to selectively retrieve and render received image data parcels to a display memory. The parcel rendering system provides the parcel request subsystem with the image parcel request of the assigned priority.

An advantage of the present invention is that both image parcel data requests and the rendering of image data are optimized to address the display based on the display resolution of the client system.

Another advantage of the present invention is that the prioritization of image parcel requests is based on an adaptable parameter that minimizes the computational complexity of determining request prioritization and, in turn, the progressive improvement in display resolution within the field of view presented on a client display.

A further advantage of the present invention is that the client software system requires relatively minimal client processing power and storage capacity. Compute intensive numerical calculations are minimally required and image parcel data is compactly stored in efficient data structures. The client software system is very small and easily downloaded to conventional computer systems or embedded in conventional dedicated function devices, including portable devices, such as PDAs and webphones.

Still another advantage of the present invention is that image parcel data requests and presentation can be readily optimized to use low to very low bandwidth network connections. The software system of the present invention provides for re-prioritization of image parcel data requests and presentation in circumstances where the rate of point-of-view navigation exceeds the data request rate.

Yet another advantage of the present invention is that image parcel data rendering is performed without requiring any complex underlying hardware or software display subsystem. The client software system of the present invention includes a bit-map rendering engine that draws directly to the video memory of the display, thus placing minimal requirements on any underlying embedded or disk operating system and display drivers. Complex graphics and animation abstraction layers are not required.

Still another advantage of the present invention is that image parcel block compression is used to obtain fixed size transmission data blocks. Image parcel data is recoverable from transmission data using a relatively simple client decompression algorithm. Using fixed size transmission data blocks enables image data parcels to be delivered to the client in bounded time frames.

A yet further advantage of the present invention is that multiple data forms can be transferred to the client software system for concurrent display. Sparse array overlay data, correlated positionally to the image parcel data and generally insensitive to image parcel resolution, can be initially or progressively provided to the client for parsing and parallel presentation on a client display image view.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other advantages and features of the present invention will become better understood upon consideration of the following detailed description of the invention when considered in connection with the accompanying drawings, in which like reference numerals designate like parts throughout the figures thereof, and wherein:

FIG. **1** depicts a preferred system environment within which various embodiments of the present invention can be utilized;

FIG. **2** is a block diagram illustrating the preparation of image parcel and overlay data set that are to be stored by and served from a network server system in accordance with a preferred embodiment of the present invention;

FIG. **3** is a block diagram of a client system image presentation system constructed in accordance with a preferred embodiment of the present invention;

FIG. **4** provides a data block diagram illustrating an optimized client image block processing path constructed in accordance with a preferred embodiment of the present invention;

FIG. **5** is a process flow diagram showing a main processing thread implemented in a preferred embodiment of the present invention;

FIG. **6** provides a process flow diagram showing a network request thread implemented in a preferred embodiment of the present invention;

FIG. **7** provides a process flow diagram showing a display image rendering thread implemented in a preferred embodiment of the present invention;

FIG. **8** provides a process flow diagram showing the parcel map processing performed preliminary to the rendering of image data parcels in accordance with a preferred embodiment of the present invention;

FIG. **9** provides a process flow diagram detailing the rendering and progressive prioritization of image parcel data download requests in accordance with a preferred embodiment of the present invention; and

FIG. **10** provides a process flow diagram detailing the determination of an optimal detail level for image parcel presentation for a current viewing frustum in accordance with a preferred embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

The preferred operational environment **10** of the present invention is generally shown in FIG. **1**. A network server system **12**, operating as a data store and server of image data, is responsive to requests received through a communications network, such as the Internet **14** generally and various tiers of internet service providers (ISPs)z including a wireless connectivity provider **16**. Client systems, including conventional workstations and personal computers **18** and smaller, typically dedicated function devices often linked through wireless network connections, such as PDAs, webphones **20**, and automobile navigation systems, source image requests to the network server **12**, provide a client display and enable image navigational input by a user of the client system. Alternately, a dedicated function client system **20** may be connected through a separate or plug-in local network server **22**, preferably implementing a small, embedded Web server, to a fixed or removable storage local image repository **24**. Characteristically, the client system **18**, **20** displays are operated at some fixed resolution generally dependent on the underlying display hardware of the client systems **18**, **20**.

The image navigation capability supported by the present invention encompasses a viewing frustum placed within a three-dimensional space over the imaged displayed on the client **18**, **20**. Client user navigational inputs are supported to control the x, y lateral, rotational and z height positioning of the viewing frustum over the image as well as the camera angle of incidence relative to the plane of the image. To effect these controls, the software implemented on the client systems **18**, **20** supports a three-dimensional transform of the image data provided from the server **12**, **22**.

In accordance with the preferred embodiments of the present invention, as generally illustrated in FIG. **2**, a network image server system **30** stores a combination of source image data **32** and source overlay data **34**. The source image data **32** is typically high-resolution bit-map satellite imagery of geographic regions, which can be obtained from commercial suppliers. The overlay image data **34** is typically a discrete data file providing image annotation information at defined coordinates relative to the source image data **32**. In the preferred embodiments of the present invention, image annotations include, for example, street, building and landmark names, as well as representative 2 and 3D objects, graphical icons, decals, line segments, and text and other characters.

The network image server system **30** preferably pre-processes the source image data **32** and source overlay data **34** to forms preferred for storage and serving by the network server **12**, **22**. The source image data **32** is preferably pre-processed to obtain a series K_{1-N} of derivative images of progressively lower image resolution. The source image data **32**, corresponding to the series image K_0, is also subdivided into a regular array such that each resulting image parcel of the array has a 64 by 64 pixel resolution where the image data has a color or bit per pixel depth of 16 bits, which represents a data parcel size of 8K bytes. The resolution of the series K_{1-N} of derivative images is preferably related to that of the source image data **32** or predecessor image in the series by a factor of four. The array subdivision is likewise related by a factor of four such that each image parcel is of a fixed 8K byte size.

In the preferred embodiment of the present invention, the image parcels are further compressed and stored by the network server **12**, **22**. The preferred compression algorithm implements a fixed 4:1 compression ratio such that each compressed and stored image parcel has a fixed 2K byte size. The image parcels are preferably stored in a file of defined configuration such that any image parcel can be located by specification of a K_D, X, Y value, representing the image set resolution index D and corresponding image array coordinate.

The source overlay data **34** is preferably pre-processed **36** into either an open XML format, such as the Geography Markup Language (GML), which is an XML based encoding standard for geographic information developed by the OpenGIS Consortium (OGC; www.opengis.org), or a proprietary binary representation. The XML/GML representation is preferred as permitting easier interchange between different commercial entities, while the binary representation is preferred as more compact and readily transferable to a client system **18**, **20**. In both cases, the source overlay data **34** is pre-processed to contain the annotation data preferably in a resolution independent form associated with a display coordinate specification relative to the source image data **32**. The XML, GML or binary overlay data may be compressed prior to storage on the network server **12**, **22**.

The preferred architecture **40** of a client system **18**, **20**, for purposes of implementing the present invention, is shown in FIG. **3**. The architecture **40** is preferably implemented by a software plug-in or application executed by the client system **18**, **20** and that utilizes basic software and hardware services provided by the client system **18**, **20**. A parcel request client **42** preferably implements an HTML client that supports HTML-based interactions with the server **12**, **22** using the underlying network protocol stack and hardware network interface provided by the client systems **18**, **20**. A central parcel processing control block **44** preferably implements the client process and control algorithms. The control block **44** directs the transfer of received image parcels and XML/GML/binary overlay data to a local parcel data store **46**. Preferably image data parcels are stored in conventional quad-tree data structures, where tree nodes of depth D correspond to the stored image parcels of a derivative image of resolution K_D. The XML/GML/binary overlay data is preferably stored as a data object that can be subsequently read by a n XML/GML/binary parser implemented as part of the control block **44**.

The control block **44** is also responsible for decompressing and directing the rendering of image parcels to a local display by a rendering engine **48**. Preferably, the rendering engine **48** writes to the video memory of the underlying client display hardware relying on only generic graphics acceleration hardware capabilities. In general, the relied on capabilities include bit-bit and related bit-oriented functions that are readily supported by current conventional display controller hardware. The rendering engine **48** is optimized to perform image parcel texture mapping without reliance on complex floating point operations, permitting even relatively simple processors to efficiently execute the rendering engine **48**.

Changes in the viewing frustum are determined from user input navigation commands by a frustum navigation block **50**. In the preferred embodiments of the present invention, the input navigation controls are modeled for three-dimensional fly-over navigation of the displayed image. The navigation controls support point-of-view rotation, translation, attitude, and altitude over the displayed image. The effective change in viewing frustum as determined by the frustum navigation block **50** is provided to the control block **44**.

The control block **44**, based in part on changes in the viewing frustum, determines the ordered priority of image parcels to be requested from the server **12**, **22** to support the progressive rendering of the displayed image. The image parcel requests are placed in a request queue **52** for issuance by the parcel request client **42**. Preferably, the pending requests are issued in priority order, thereby dynamically reflecting changes in the viewing frustum with minimum latency.

An optimal image parcel data flow **60**, as configured for use in the preferred embodiments of the present invention, is shown in FIG. **4**. Preferably, the TCP/IP network protocol is used to deliver image parcels to the clients **18**, **20**. For the preferred embodiment., where network bandwidth is limited or very limited, entire image parcels are preferably delivered in corresponding data packets. This preference maximizes data delivery while avoiding the substantial latency and processing overhead of managing image parcel data split over multiple network packets. Thus, a 2K byte compressed image parcel **62** is delivered as the data payload of a TCP/IP packet **64**. Uncompressed, the 8K byte image parcel **62** is recognized as part of the present invention as being within the nominally smallest L1 data cache **66** size of conventional microprocessors **68**. By ensuring that an uncompressed image parcel fits within the L1 cache, the texture map rendering algorithm can execute with minimum memory management overhead, thus optimally utilizing the processing capability of the microprocessor **68**. Additionally, the writing of video data as a product of the rendering algorithm is uniform, thereby improving the apparent video stability of the display to the user.

The client architecture **40** preferably executes in multiple process threads, with additional threads being utilized for individual network data request transactions. As shown in FIG. **5**, an image parcel management process **80** implements a loop that determines image parcels subject to update **82** and creates corresponding image parcel download requests **84**. Navigation events that alter the viewing frustum are considered in part to determine the current field of view. The quad-tree data structures are examined **86** to identify viewable image parcels of higher resolution than currently available in the parcel data store **46**.

A pool of image request threads is preferably utilized to manage the image parcel download operations. In the preferred embodiments of the present invention, a pool of four network request threads is utilized. The number of pool threads is determined as a balance between the available system resources and the network response latency, given the available bandwidth of the network connection. Empirically, for many wireless devices, four concurrent threads are able to support a relatively continuous delivery of image data parcels to the client **20** for display processing. As image parcels are progressively identified for download, a free request thread is employed to issue **88** a corresponding network request to the server **12**, **22**. When a network response is received, the corresponding thread recovers **90** the image parcel data. The received image parcel is then stored **92** in a corresponding quad-tree data structure node.

For small clients **20**, the available memory for the parcel data store **46** is generally quite restricted. In order to make optimal use of the available memory, only currently viewable image parcels are subject to download. Where the size of the parcel data store **46** is not so restricted, this constraint can be relaxed. In either case, a memory management process **94** runs to monitor use of the parcel data store **46** and selectively remove image parcels to free memory for newly requested image parcels. Preferably, the memory management process **94** operates to preferentially remove image parcels that are the furthest from the current viewing frustum and that have the highest data structure depth. Child node image parcels are always removed before a parent node parcel is removed.

A preferred network request management process **100** is shown in FIG. **6**. The process **100** waits **102** on the existence of a download request in the priority request queue **52**. The process **100** then waits on a network request pool thread to become free **104**. When a network request thread becomes available, the process **100** examines **106** all of the pending requests in the priority request queue **52** and selects **108** the request with the highest assigned priority. Thus, sequentially enqueued requests can be selectively issued out of order based on an independently assigned request priority. The request is then issued **110** and the request management process **100** leaves the request thread waiting on a network response.

FIG. **7** presents a preferred display management process **120**. Event driven user navigation information is evaluated **122** to determine a current viewing frustum location and orientation within a three-dimensional space relative to the displayed image. An algorithmic priority selection **124** of a next image parcel to render is then performed. The selected image parcel is then rendered **126** to the display memory **70**. The rendering operation preferably performs a texture map transform of the parcel data corresponding to the current viewing frustum location and orientation. The overlay data is then parsed or is pre-parsed to determine **128** whether the image coordinates of any overlay annotation correspond to the current image parcel location. If the coordinates match, the overlay annotation is rendered **130** to the video display memory **70**. The process **120** then continues with the next selection **124** of an image parcel to render, subject to any change in the viewing frustum location and orientation.

A preferred implementation of the selection **124** and rendering **126** of image parcels in accordance with the present invention is detailed in FIGS. **8** through **10**. Referring first to FIG. **8**, any outstanding requests in the priority request queue **52** are preferably cleared **142** in response to a change in the viewing frustum location and orientation. The effective altitude of the viewing frustum and the resolution of the client display are then used as a basis for determining an optimal level of detail L that will be displayed. The detail level L value operates as a floor defining the maximum resolution K_L of image data that can be effectively viewed on the client display given the location and orientation of the viewing frustum. Constraining image parcel requests to the resolution range K_N to K_L, where K_N

is the lowest resolution derivative image stored by the network server **12**, **22**, prevents the download and processing of image parcels that cannot provide any perceptible improvement in the displayed image.

As part of the recursive evaluation of the optimal level of detail L, the image display space is progressively split **146** by four to one reductions into polygons. The quad-tree data structures holding existing image parcel data in the parcel data store **46** are concurrently traced **148** to establish a correspondence with the polygon map. Where the trace of a quad-tree data structure completes **150** to a node index of L for a polygon P, the node corresponding image parcel is associated with polygon P. The polygon P will not be further subdivided and no higher resolution image parcels will be requested for any portion of the image within the area represented by polygon P. Where the trace reaches a maximum node index of D for a polygon P' **152**, where $N \leqq D < L$ and N is the index of the lowest resolution derivative image stored by the network server **12**, **22**, the image parcel associated with the node is associated with the polygon P'. This polygon P' will be subject to further subdivision and progressive requests for image parcels of higher resolution up to the detail level L.

Referring now to FIG. **9**, a display image is then rendered **160** beginning with the maximum depth polygons previously found. Iterating over the set of maximum depth polygons, any polygons outside of the viewing frustum are skipped **162**. Polygons that are at least partially visible are clipped to the applicable bounds of the viewing frustum **164**. The polygon corresponding image parcel data is then texture mapped **166** into the polygon corresponding coordinates of the video memory **70**. If the node index depth of the rendered image parcel is at least equal to the prior determined optimal detail level L **168**, the iteration over the polygons P continues.

Where the node index depth is less than the optimal detail level L **170**, the polygon P' is subdivided into four polygons and correspondingly represented by the creation of four child nodes within the associated quad-tree data structure **172**. Four image parcel download requests are then created **174**.

The download priority associated with each request is determined **176** by execution of a function S that operates on a 2D polygon argument P and returns a real number representing the request priority. The function argument P is a list of real (x, y) coordinates of the vertices of the current polygon in screen coordinates after being clipped to fit within the current viewing frustum. That is, the function S works over general polygons in a two-dimensional space, whose vertices are specified by the series $\{(x(1),y(1)),(x(2),y(2)), \ldots ,(x(n),y(n))\}$. The argument P vertices sent to S represent the position of the vertices composing each of the polygons, after being clipping to the viewing frustum, viewable within the display space having the fixed resolution [xRes, yRes]. Thus, the clipped polygons are all within the rectangle $[0, xRes] \times [0, yRes]$.

In execution of the function S, each of the P coordinates is first transformed by linear mapping of the screen coordinate space to the square $[-1,1] \times [-1,1]$ by the operation $x(i):=(x(i)-xRes/2)/(xRes/2)$; $y(i):=(y(i)-yRes/2)/(yRes/2)$. The x and y coordinate values of each vertex (x(i),y(i)) for i=1 to n) are then transformed by the function $T(a)=sgn(a)*pow(|a|, d)$, where the control parameter d is a constant in the range (0,1], or equivalently the interval $0<d\leqq 1$. The function S then returns a real value that is equal to the area covered by the argument polygon P vertices subject to the applied coordinate transformation. Thus, the accumulated priority for any image parcel pending download is the sum of the values of returned by the function S for each of the viewable polygons that require some part of the image parcel as the source data for texture mop rendering of the polygon. The priority operation of the request queue **52** is such that download requests will be issued preferentially for image parcels with the largest priority value.

In accordance with the preferred embodiments of the present invention, the value of the control parameter d can be adjusted to ultimately affect the behavior of the function S in determining the download request priority. In general, image parcels with lower resolution levels will accumulate greater priority values due to the larger number of polygons that may use a given low resolution image parcel as a rendering data source. Such lower resolution image parcels are therefore more likely to be preferentially downloaded. In accordance with the present invention, this generally assures that a complete image of at least low resolution will be available for rendering.

The control parameter d, as applied in execution of the function S, well as the area distortion produced by the projection transform also influences the value returned by the function S such that relatively higher-resolution image parcels near the image view point will occasionally achieve a higher priority than relatively remote and partially viewed image parcels of lower resolution. Using values smaller than 1 for the control parameter d results in requests with a higher priority for parcels covering areas near the focal point of the viewer, which is presumed to be the center point of the display space, relative to requests for parcels further from the center point in absolute terms and of the same resolution depth D. Thus, in accordance with the present invention, the priority assigned to image parcel requests effectively influences the order of requests based on the relative contribution of the image parcel data to the total display quality of the image. Empirically, a value of 0.35 for the control parameter d for small screen devices, such as PDAs and webphones has been found to produce desirable results.

The computed priorities of each of the four newly created image parcel requests are then assigned **178** and the requests are enqueued in the priority request queue **52**. The next polygon P is then considered in the loop of the image parcel rendering process **160**.

The preferred algorithm **180** for determining the detail level L value for a given viewing frustum is shown in FIG. **10**. In accordance with the present invention, the optimal detail level L is effectively the limit at which the resolution of image parcel data functionally exceeds the resolution of the client display. Preferably, to determine the optimal detail level L, the viewpoint or camera position of the viewing frustum is determined **182** relative to the displayed image. A nearest polygon P of depth D is then determined **184** from the effective altitude and attitude of the viewpoint. The nearest point A of the polygon P is then determined **186**. The point A may be within the interior or an edge of the polygon P, though most likely be located at a vertex of the polygon P.

The optimum level of detail L at point A is then computed **188** as the base-4 logarithm of the number of pixels on the screen that would be covered by a single pixel from an image parcel of the lowest resolution K_{-N} image, which is the quad-tree root image and corresponds to an image area covering the entire image map. The point A optimal detail level L is preferably computed analytically from the local value of the Jacobian of the projective transform used to transform the three dimensional image coordinate space to screen coordinates, evaluated at the point A.

Where the depth D of the polygon P is greater than the depth of the computed optimal level of detail L, the detail level L is taken as the optimal detail level L **190**. Thus, through the process **140**, an image parcel or corresponding section of the closest resolution image parcel associated with a parent node in the quad-tree data structure relative to the depth level L will be used as the texture for rendering the polygon P. Conversely, if the depth D is less than that of the optimal detail level L, the polygon P is effectively split into quadrants and the optimal level of detail is reevaluated. The process **180** thus continues iteratively until the optimal detail level L is found.

Thus, a system and methods of optimally presenting image data on client systems with potentially limited processing performance, resources, and communications bandwidth have been described. While the present invention has been described particularly with reference to the communications and display of geographic image data, the present invention is equally applicable to the efficient communications and display of other high resolution information.

In view of the above description of the preferred embodiments of the present invention, many modifications and variations of the disclosed embodiments will be readily appreciated by those of skill in the art. It is therefore to be understood that, within the scope of the appended claims, the invention may be practiced otherwise than as specifically described above.

The invention claimed is:

1. A client system for dynamic visualization of image data provided through a network communications channel, said client system comprising:

a parcel request subsystem, including a parcel request queue, operative to request discrete image data parcels in a priority order and to store received image data parcels in a parcel data store, said parcel request subsystem being responsive to an image parcel request of assigned priority to place said image parcel request in said parcel request queue ordered in correspondence with said assigned priority;

an parcel rendering subsystem coupled to said parcel data store to selectively retrieve and render received image data parcels to a display memory, said parcel rendering system providing said parcel request subsystem with said image parcel request of said assigned priority;

wherein said parcel rendering subsystem determines said assigned priority based on a determined optimal image resolution level;

wherein said display memory is coupled to an image display of predetermined resolution and wherein said determined optimal image resolution level is based on said predetermined resolution;

wherein said assigned priority further reflects the proximity of the image parcel referenced by said image parcel request to a predetermined focal point;

wherein said discrete image data parcels are of a first fixed size as received by said parcel request subsystem and of a second fixed size as rendered by said parcel rendering subsystem; and

wherein said discrete image data parcels each includes a fixed-size array of pixel data.

2. A method of supporting dynamic visualization of image data transferred through a communications channel, said method comprising the steps of:

determining, in response to user navigational commands, a viewpoint orientation with respect to an image displayed within a three-dimensional space;

requesting, in a priority order, image parcels renderable as corresponding regions of said image, each said image parcel having an associated resolution, wherein said priority order is determined to provide a progressive regional resolution enhancement of said image as each said image parcel is rendered;

receiving a plurality of image parcels through said communications channel;

rendering said plurality of image parcels to provide said image;

wherein said step of receiving includes the step of storing said plurality of image parcels in an image store and wherein said step of rendering provides for the selective rendering of said plurality of image parcels having the highest associated resolutions to the corresponding regions of said image;

wherein said step of rendering limits the selective rendering of said image parcels to image parcels having associated resolutions less than a predetermined level;

wherein said step of rendering selectively renders said plurality of image parcels as the unique textures for the corresponding regions of said image; and

wherein said priority order is re-evaluated in response to a chance in said viewpoint orientation.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 7,139,794 B2 Page 1 of 1
APPLICATION NO. : 10/035981
DATED : November 21, 2006
INVENTOR(S) : Isaac Levanon and Yoni Lavi

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

In the Specification:

Column 1, line 8: change “60/258,485” to --60/258,489--

In the Claims:

Column 12, line 45: change “chance” to --change--

Signed and Sealed this
First Day of July, 2014

Michelle K. Lee

Michelle K. Lee
Deputy Director of the United States Patent and Trademark Office

EXHIBIT D

To the First Amended Complaint

US007908343B2

(12) **United States Patent**
Levanon et al.

(10) **Patent No.: US 7,908,343 B2**
(45) **Date of Patent: *Mar. 15, 2011**

(54) **OPTIMIZED IMAGE DELIVERY OVER LIMITED BANDWIDTH COMMUNICATION CHANNELS**

(75) Inventors: **Isaac Levanon**, Ramat Hasharn (IL); **Yoni Lavi**, Raanana (IL)

(73) Assignee: **Inovo Limited**, The Valley Anguilla (VG)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **12/619,643**

(22) Filed: **Nov. 16, 2009**

(65) **Prior Publication Data**

US 2010/0064002 A1 Mar. 11, 2010

Related U.S. Application Data

(63) Continuation of application No. 10/035,987, filed on Dec. 24, 2001, now Pat. No. 7,644,131.

(60) Provisional application No. 60/258,488, filed on Dec. 27, 2000, provisional application No. 60/258,489, filed on Dec. 27, 2000, provisional application No. 60/258,465, filed on Dec. 27, 2000, provisional application No. 60/258,468, filed on Dec. 27, 2000, provisional application No. 60/258,466, filed on Dec. 27, 2000, provisional application No. 60/258,467, filed on Dec. 27, 2000.

(51) **Int. Cl.**
G06F 15/16 (2006.01)

(52) **U.S. Cl.** **709/217**; 709/230; 345/625; 382/232; 382/305

(58) **Field of Classification Search** 709/202, 709/203, 217, 230, 246, 247; 345/625; 382/232, 382/305
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,929,860	A	7/1999	Hoppe	
6,182,114	B1	1/2001	Yap et al.	
6,212,301	B1	4/2001	Warner et al.	
6,314,452	B1	11/2001	Dekel et al.	
6,326,965	B1	12/2001	Castelli et al.	
6,345,279	B1	2/2002	Li et al.	
6,346,938	B1	2/2002	Chan et al.	
6,397,259	B1	5/2002	Lincke et al.	
6,608,933	B1	8/2003	Dowell et al.	
6,704,024	B2	3/2004	Robotham et al.	
6,711,297	B1	3/2004	Chang et al.	
6,801,665	B1	10/2004	Atsumi et al.	
7,644,131	B2 *	1/2010	Levanon et al.	709/217

OTHER PUBLICATIONS

Hoppe, H. "Progressive Meshes", SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, 1996, pp. 99-108.

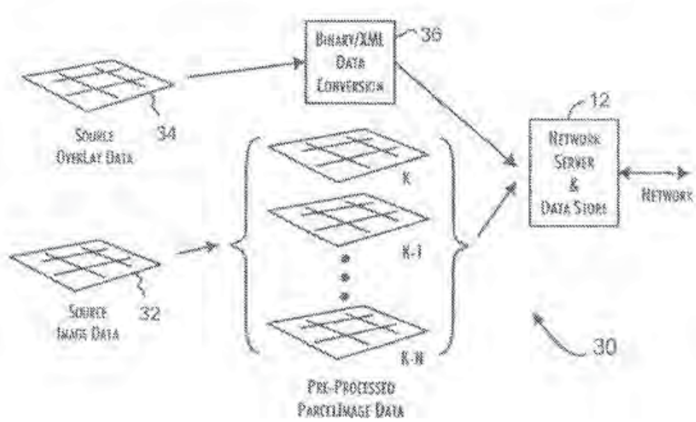
* cited by examiner

Primary Examiner — David Lazaro

(57) **ABSTRACT**

Large-scale images are retrieved over network communications channels for display on a client device by selecting an update image parcel relative to an operator controlled image viewpoint to display via the client device. A request is prepared for the update image parcel and associated with a request queue for subsequent issuance over a communications channel. The update image parcel is received from the communications channel and displayed as a discrete portion of the predetermined image. The update image parcel optimally has a fixed pixel array size, is received in a single and or plurality of network data packets, and is constrained to a resolution less than or equal to the resolution of the client device display.

20 Claims, 5 Drawing Sheets



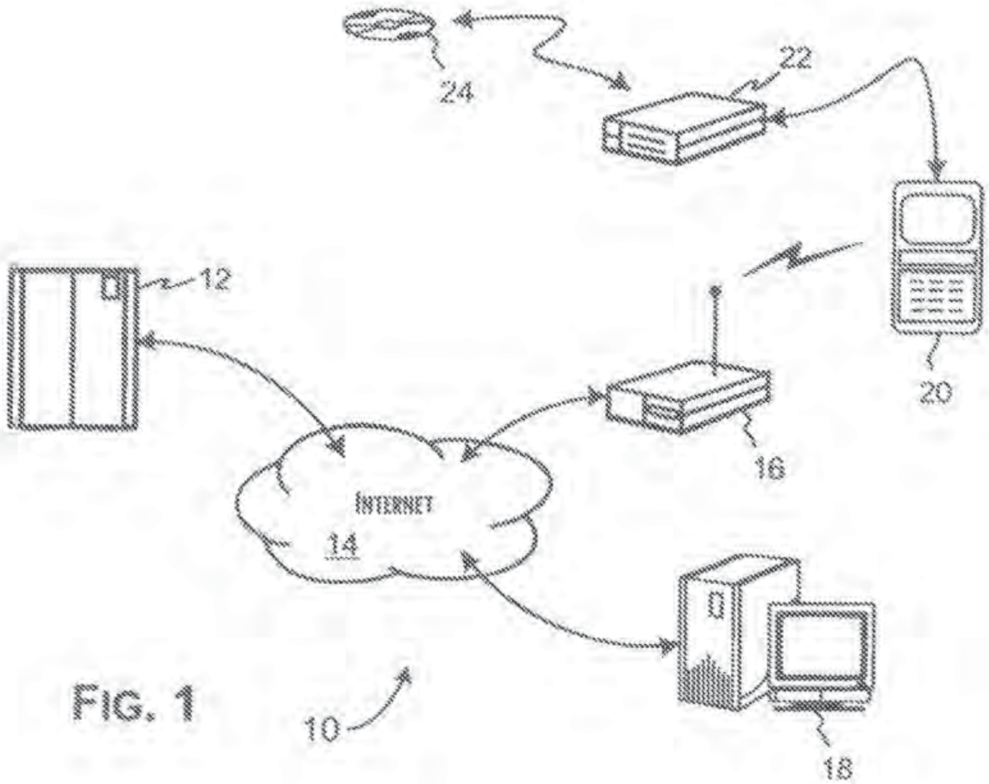

24
22
12
20
16
INTERNET
14
10
18

FIG. 1

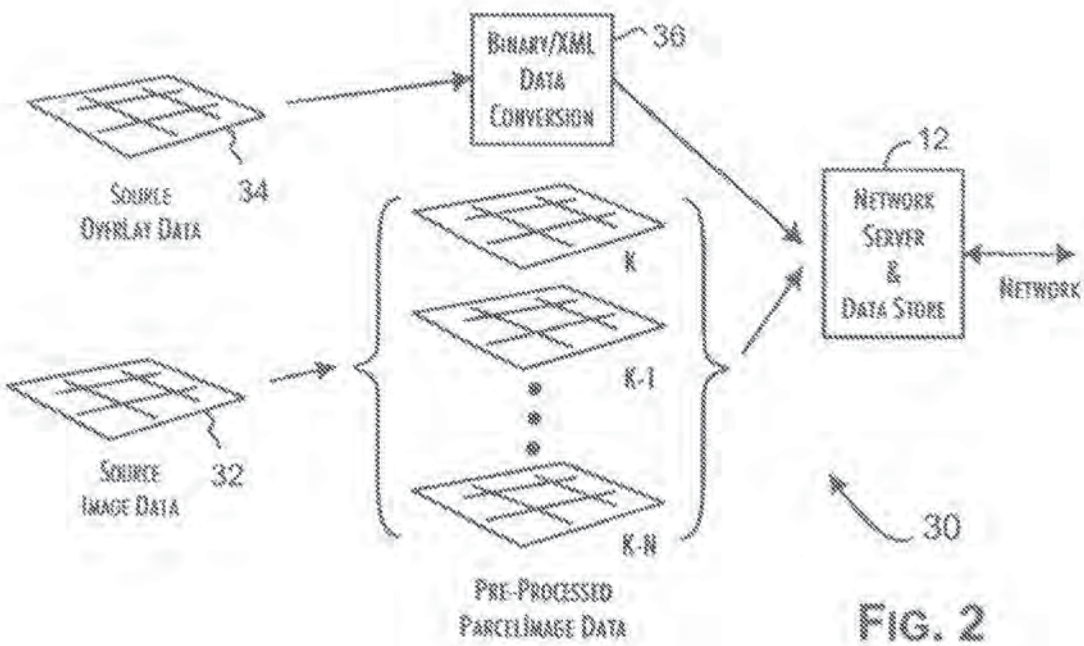

BINARY/XML
DATA
CONVERSION
36
SOURCE
OVERLAY DATA
34
12
NETWORK
SERVER
&
DATA STORE
NETWORK
K
K-1
K-N
SOURCE
IMAGE DATA
32
PRE-PROCESSED
PARCELIMAGE DATA
30

FIG. 2

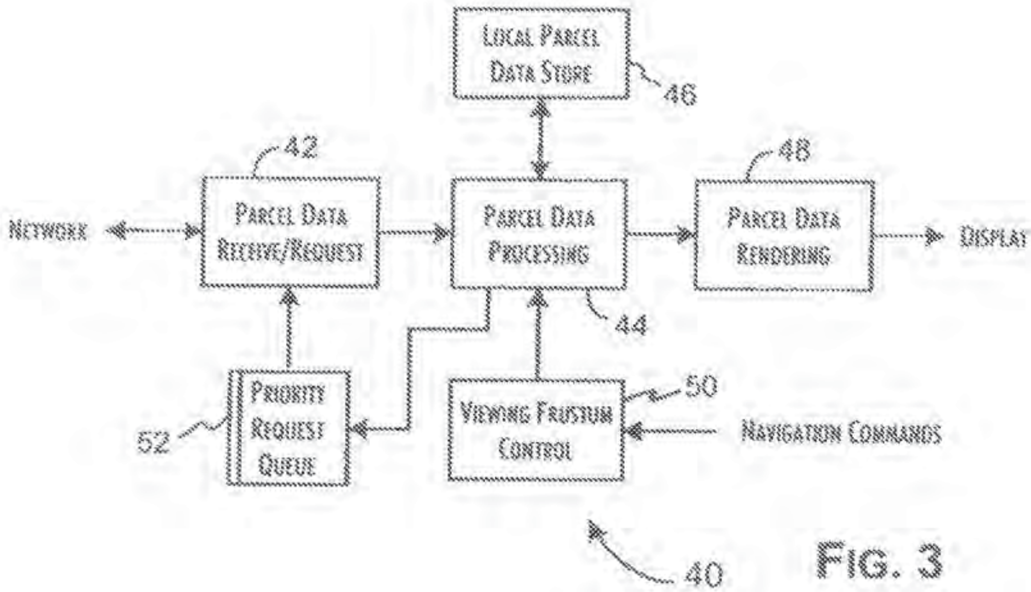

Local Parcel Data Store
46
42
48
Network
Parcel Data Receive/Request
Parcel Data Processing
Parcel Data Rendering
Display
44
50
52
Priority Request Queue
Viewing Frustum Control
Navigation Commands
40

FIG. 3

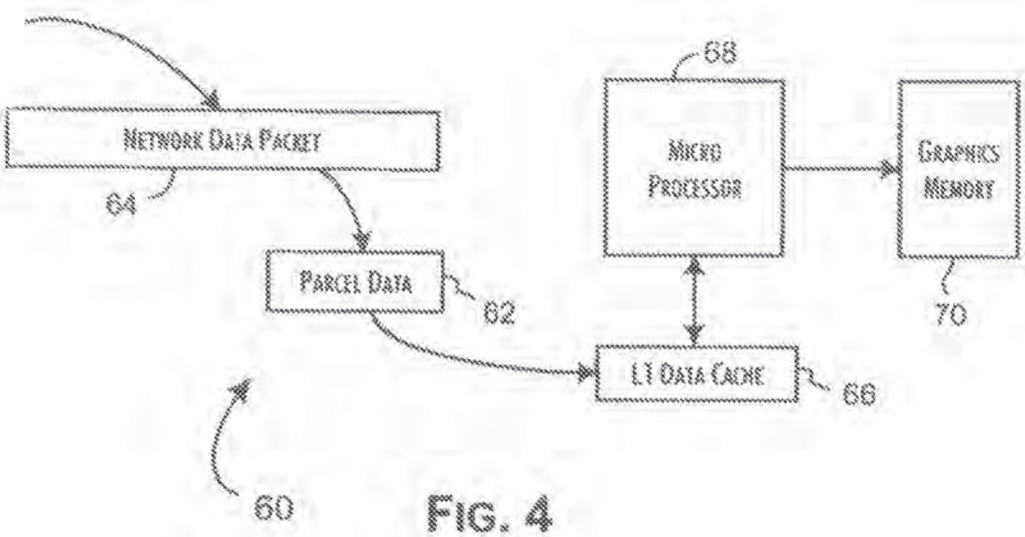

68
Network Data Packet
Micro Processor
Graphics Memory
64
Parcel Data
62
70
L1 Data Cache
66
60

FIG. 4

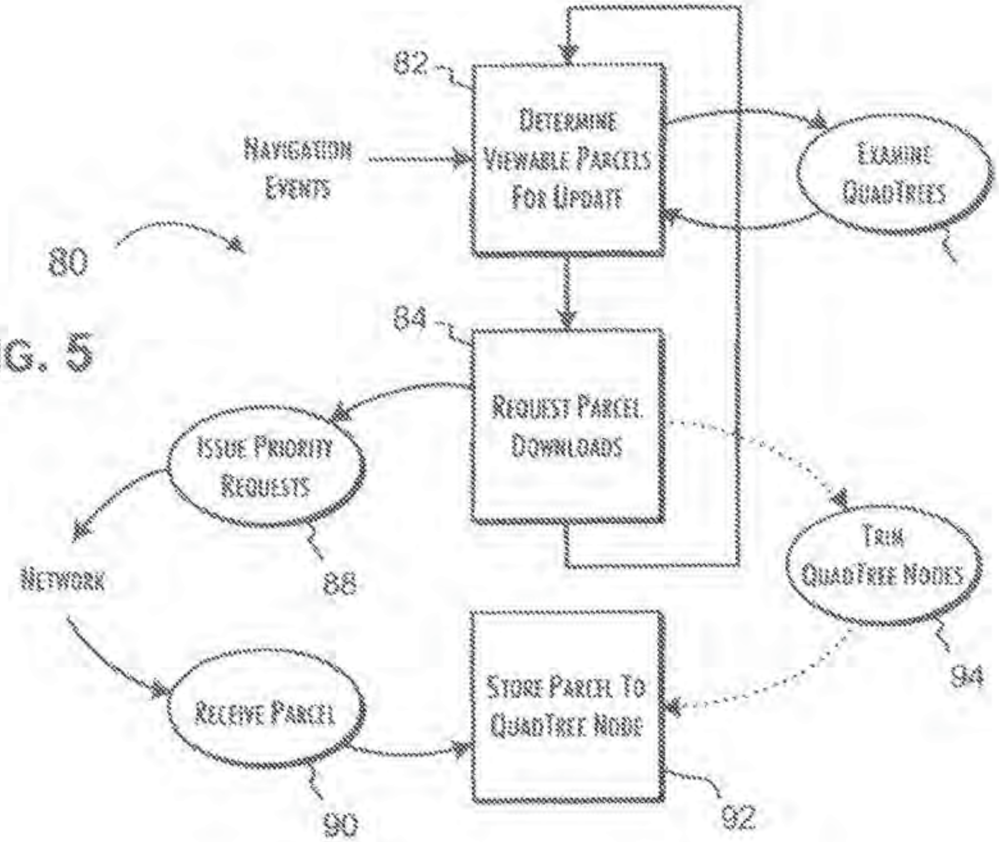

82
Navigation Events
Determine Viewable Parcels For Update
Examine QuadTrees
80
84
Request Parcel Downloads
Issue Priority Requests
Network
88
Trim QuadTree Nodes
94
Receive Parcel
Store Parcel To QuadTree Node
90
92

FIG. 5

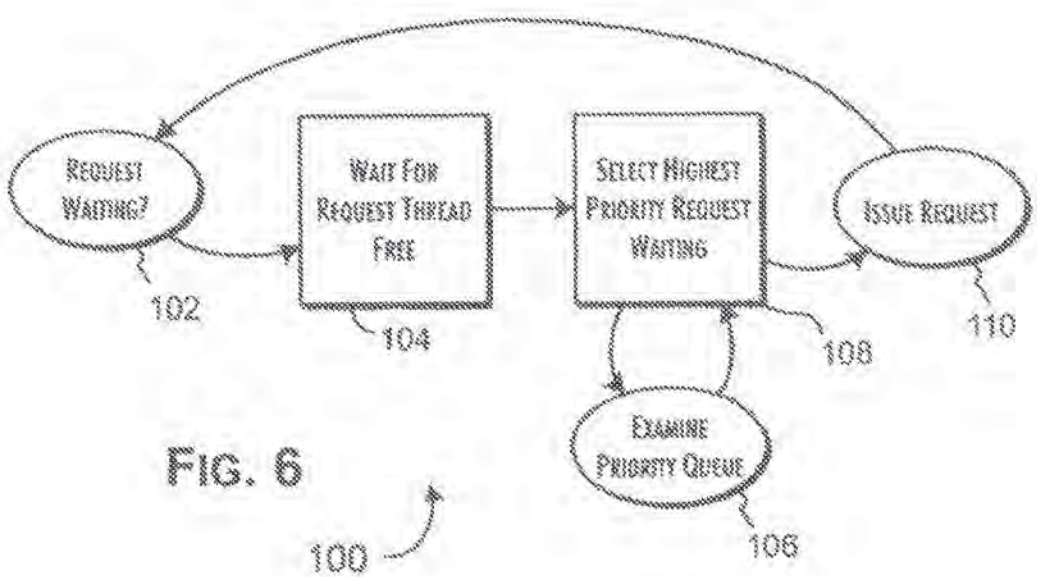

Request Waiting?
Wait For Request Thread Free
Select Highest Priority Request Waiting
Issue Request
102
104
108
110
Examine Priority Queue
106
100

FIG. 6

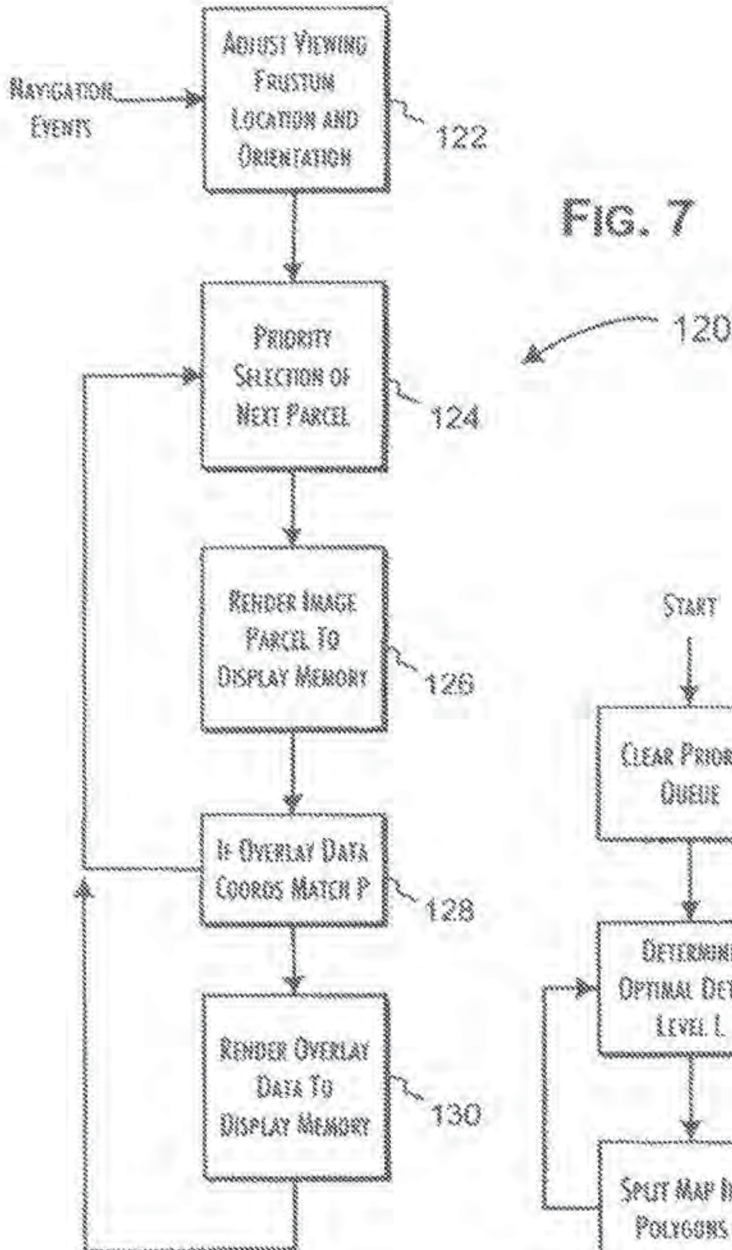

Navigation Events
Adjust Viewing Frustum Location and Orientation
122
Priority Selection of Next Parcel
124
Render Image Parcel To Display Memory
126
If Overlay Data Coords Match P
128
Render Overlay Data To Display Memory
130
120

FIG. 7

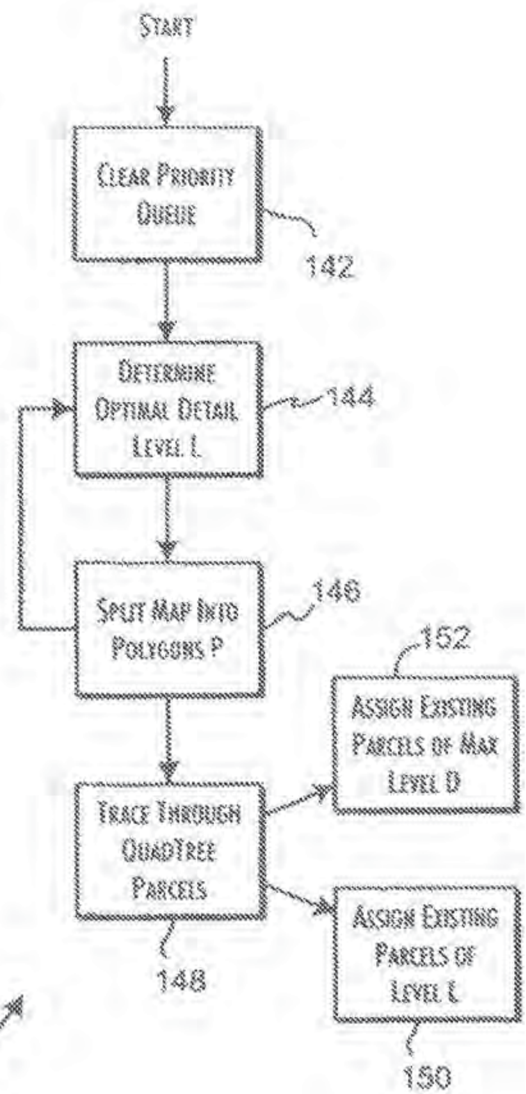

Start
Clear Priority Queue
142
Determine Optimal Detail Level L
144
Split Map Into Polygons P
146
Trace Through QuadTree Parcels
148
Assign Existing Parcels of Max Level D
152
Assign Existing Parcels of Level L
150
140

FIG. 8

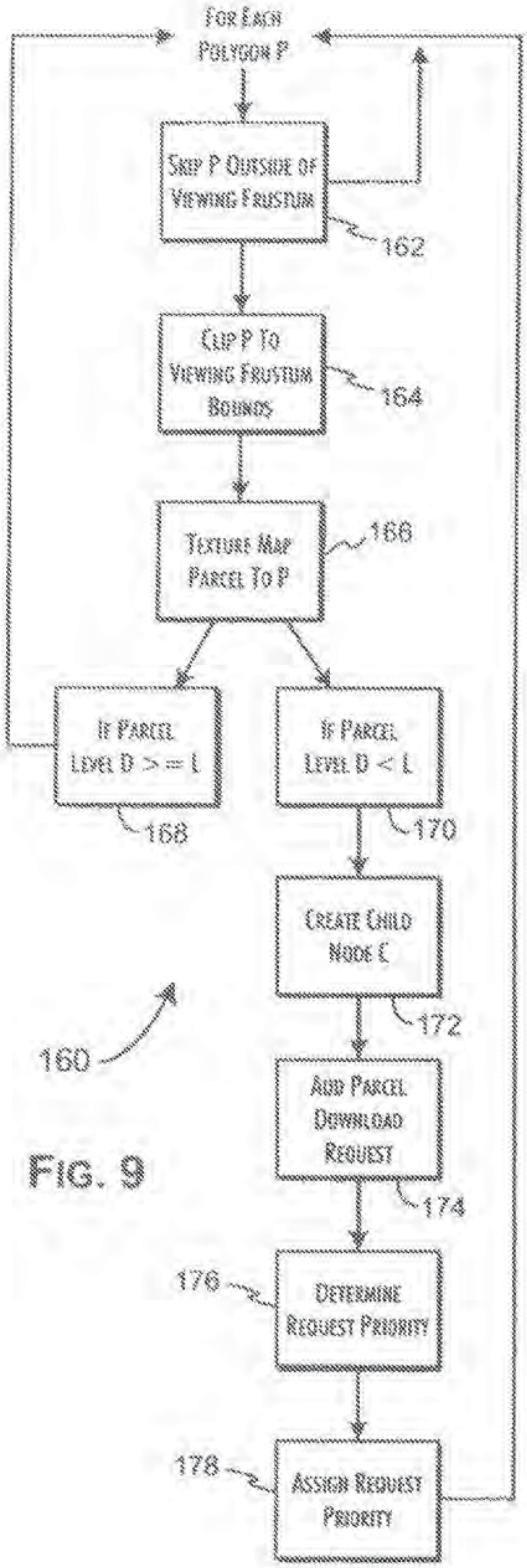

For Each Polygon P
Skip P Outside of Viewing Frustum
162
Clip P To Viewing Frustum Bounds
164
Texture Map Parcel To P
166
If Parcel Level D >= L
168
If Parcel Level D < L
170
Create Child Node C
172
Add Parcel Download Request
174
176
Determine Request Priority
178
Assign Request Priority
160

FIG. 9

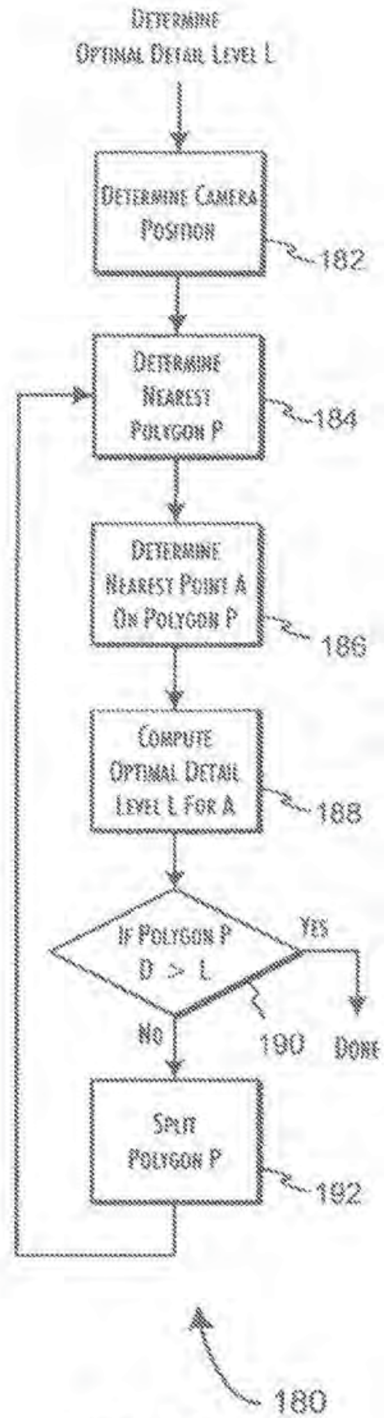

Determine Optimal Detail Level L
Determine Camera Position
182
Determine Nearest Polygon P
184
Determine Nearest Point A On Polygon P
186
Compute Optimal Detail Level L For A
188
If Polygon P D > L
Yes
No
190
Done
Split Polygon P
192
180

FIG. 10

OPTIMIZED IMAGE DELIVERY OVER LIMITED BANDWIDTH COMMUNICATION CHANNELS

PRIORITY CLAIMS/RELATED APPLICATIONS

This application is a continuation of and claims priority under 35 USC 120 to U.S. patent application Ser. No. 10/035,987 filed on Dec. 24, 2001 and entitled "Optimized image delivery over limited bandwidth communication channels" which in turn claims the benefit under 35 USC 119(e) of U.S. Provisional Application Nos. 60/258,488, 60/258,489, 60/258,465, 60/258,468, 60/258,466, and 60/258,467, all filed Dec. 27, 2000, all of which are incorporated herein by reference. The present application is also related to the co-pending application Ser. No. 10/035,981 entitled "System and Methods for Network Image Delivery with Dynamic Viewing Frustum Optimized for Limited Bandwidth Communication Channels, Levanon et al., filed on Dec. 24, 2001 (now U.S. Pat. No. 7,139,794 issued on Nov. 21, 2006 and which is assigned to the Assignee of the present Application.

FIELD

The disclosure is related to network based, image distribution systems and, in particular, to a system and methods for efficiently selecting and distributing image parcels through a narrowband or otherwise limited bandwidth communications channel to support presentation of high-resolution images subject to dynamic viewing frustums.

BACKGROUND

The Internet and or other network systems may provide a unique opportunity to transmit for example complex images, typically large scale bit-maps, particularly those approaching photo-realistic levels, over large area and or distances. In common application, the images may be geographic, topographic, and or other highly detailed maps. The data storage requirements and often proprietary nature of such images could be such that conventional interests may be to transfer the images on an as-needed basis.

In conventional fixed-site applications, the image data may be transferred over a relatively high-bandwidth network to client computer systems that in turn, may render the image. Client systems may typically implement a local image navigation system to provide zoom and or pan functions based on user interaction. As well recognized problem with such conventional systems could be that full resolution image presentation may be subject to the inherent transfer latency of the network. Different conventional systems have been proposed to reduce the latency affect by transmitting the image in highly compressed formats that support progressive resolution build-up of the image within the current client field of view. Using a transform compressed image transfer function increases the field of the image that can be transferred over a fixed bandwidth network in unit time. Progressive image resolution transmission, typically using a differential resolution method, permits an approximate image to be quickly presented with image details being continuously added over time.

Tzou, in U.S. Pat. No. 4,698,689, describes a two-dimensional data transform system that supports transmission of differential coefficients to represent an image. Subsequent transmitted coefficient sets are progressively accumulated with prior transmitted sets to provide a succeedingly refined image. The inverse-transform function performed by the client computer is, however, highly compute intensive. In order to simplify the transform implementation and further reduce the latency of presenting any portion of an approximate image, images are sub-divided into a regular array. This enables the inverse-transform function on the client, which is time-critical, to deal with substantially smaller coefficient data sets. The array size in Tzou is fixed, which leads to progressively larger coefficient data sets as the detail level of the image increases. Consequently, there is an inherently increasing latency in resolving finer levels of detail.

An image visualization system proposed by Yap et al., U.S. Pat. No. 6,182,114, overcomes some of the foregoing problems. The Yap et al. system also employs a progressive encoding transform to compress the image transfer stream. The transform also operates on a subdivided image, but the division is indexed to the encoding level of the transform. The encoded transform coefficient data sets are, therefore, of constant size, which supports a modest improvement in the algorithmic performance of the inverse transform operation required on the client.

Yap et al. adds utilization of client image panning or other image pointing input information to support a foveation-based operator to influence the retrieval order of the subdivided image blocks. This two-dimensional navigation information is used to identify a foveal region that is presumed to be the gaze point of a client system user. The foveation operator defines the corresponding image block as the center point of an ordered retrieval of coefficient sets representing a variable resolution image. The gaze point image block represents the area of highest image resolution, with resolution reduction as a function of distance from the gaze point determined by the foveation operator. This technique thus progressively builds image resolution at the gaze point and succeedingly outward based on a relatively compute intensive function. Shifts in the gaze point can be responded to with relative speed by preferentially retrieving coefficient sets at and near the new foveal region.

Significant problems remain in permitting the convenient and effective use of complex images by many different types of client systems, even with the improvements provided by the various conventional systems. In particular, the implementation of conventional image visualization systems is generally unworkable for smaller, often dedicated or embedded, clients where use of image visualization would clearly be beneficial. Conventional approaches effectively presume that client systems have an excess of computing performance, memory and storage. Small clients, however, typically have restricted performance processors with possibly no dedicated floating-point support, little general purpose memory, and extremely limited persistent storage capabilities, particularly relative to common image sizes. A mobile computing device such as mobile phone, smart phone, and or personal digital assistant (PDA) is a characteristic small client. Embedded, low-cost kiosk and or automobile navigation systems are other typical examples. Such systems are not readily capable, if at all, of performing complex, compute-intensive Fourier or wavelet transforms, particularly within a highly restricted memory address space.

As a consequence of the presumption that the client is a substantial computing system, conventional image visualization systems also presume that the client is supported by a complete operating system. Indeed, many expect and require an extensive set of graphics abstraction layers to be provided by the client system to support the presentation of the delivered image data. In general, these abstraction layers are conventionally considered required to handle the mapping of the image data resolution to the display resolution capabilities of the client system. That is, resolution resolved image data provided to the client is unconstrained by any limitation in the client system to actually display the corresponding image. Consequently, substantial processor performance and memory can be conventionally devoted to handling image data that is not or cannot be displayed.

Another problem is that small clients are generally constrained to generally to very limited network bandwidths, particularly when operating under wireless conditions. Such limited bandwidth conditions may exist due to either the direct technological constraints dictated by the use of a low bandwidth data channel or indirect constraints imposed on relatively high-bandwidth channels by high concurrent user loads. Cellular connected PDAs and webphones are examples of small clients that are frequently constrained by limited bandwidth conditions. The conventionally realizable maximum network transmission bandwidth for such small devices may range from below one kilobit per second to several tens of kilobits per second. While Yap et al. states that the described system can work over low bandwidth lines, little more than utilizing wavelet-based data compression is advanced as permitting effective operation at low communications bandwidths. While reducing the amount of data that must be carried from the server to the client is significant, Yap et al. simply relies on the data packet transfer protocols to provide for an efficient transfer of the compressed image data. Reliable transport protocols, however, merely mask packet losses and the resultant, sometimes extended, recovery latencies. When such covered errors occur, however, the aggregate bandwidth of the connection is reduced and the client system can stall waiting for further image data to process.

Consequently, there remains a need for an image visualization system that can support small client systems, place few requirements on the supporting client hardware and software resources, and efficiently utilize low to very low bandwidth network connections.

SUMMARY

Thus, a general purpose of the present invention is to provide an efficient system and methods of optimally presenting image data on client systems with potentially limited processing performance, resources, and communications bandwidth.

This is achieved in the present invention by providing for the retrieval of large-scale images over network communications channels for display on a client device by selecting an update image parcel relative to an operator controlled image viewpoint to display via the client device. A request is prepared for the update image parcel and associated with a request queue for subsequent issuance over a communications channel. The update image parcel is received from the communications channel and displayed as a discrete portion of the predetermined image. The update image parcel optimally has a fixed pixel array size, is received in a single and or plurality of network data packets, and may be constrained to a resolution less than or equal to the resolution of the client device display.

An advantage of the present invention is that both image parcel data requests and the rendering of image data are optimized to address the display based on the display resolution of the client system.

Another advantage of the present invention is that the prioritization of image parcel requests is based on an adaptable parameter that minimizes the computational complexity of determining request prioritization and, in turn, the progressive improvement in display resolution within the field of view presented on a client display.

A further advantage of the present invention is that the client software system requires relatively minimal client processing power and storage capacity. Compute intensive numerical calculations are minimally required and image parcel data is compactly stored in efficient data structures. The client software system is very small and easily downloaded to conventional computer systems or embedded in conventional dedicated function devices, including portable devices, such as PDAs and webphones.

Still another advantage of the present invention is that image parcel data requests and presentation can be readily optimized to use low to very low bandwidth network connections. The software system of the present invention provides for re-prioritization of image parcel data requests and presentation in circumstances where the rate of point-of-view navigation exceeds the data request rate.

Yet another advantage of the present invention is that image parcel data rendering is performed without requiring any complex underlying hardware or software display subsystem. The client software system of the present invention includes a bit-map rendering engine that draws directly to the video memory of the display, thus placing minimal requirements on any underlying embedded or disk operating system and display drivers. Complex graphics and animation abstraction layers are not required.

Still another advantage of the present invention is that image parcel block compression is used to obtain fixed size transmission data blocks. Image parcel data is recoverable from transmission data using a relatively simple client decompression algorithm. Using fixed size transmission data blocks enables image data parcels to be delivered to the client in bounded time frames.

A yet further advantage of the present invention is that multiple data forms can be transferred to the client software system for concurrent display. Sparse array overlay data, correlated positionally to the image parcel data and generally insensitive to image parcel resolution, can be initially or progressively provided to the client for parsing and parallel presentation on a client display image view.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other advantages and features of the present invention will become better understood upon consideration of the following detailed description of the invention when considered in connection with the accompanying drawings, in which like reference numerals designate like parts throughout the figures thereof, and wherein:

FIG. **1** depicts a preferred system environment within which various embodiments of the present invention can be utilized;

FIG. **2** is a block diagram illustrating the preparation of image parcel and overlay data set that are to be stored by and served from a network server system in accordance with a preferred embodiment of the present invention;

FIG. **3** is a block diagram of a client system image presentation system constructed in accordance with a preferred embodiment of the present invention;

FIG. **4** provides a data block diagram illustrating an optimized client image block processing path constructed in accordance with a preferred embodiment of the present invention;

FIG. **5** is a process flow diagram showing a main processing thread implemented in a preferred embodiment of the present invention;

FIG. **6** provides a process flow diagram showing a network request thread implemented in a preferred embodiment of the present invention;

FIG. **7** provides a process flow diagram showing a display image rendering thread implemented in a preferred embodiment of the present invention;

FIG. **8** provides a process flow diagram showing the parcel map processing performed preliminary to the rendering of image data parcels in accordance with a preferred embodiment of the present invention;

FIG. **9** provides a process flow diagram detailing the rendering and progressive prioritization of image parcel data download requests in accordance with a preferred embodiment of the present invention; and

FIG. **10** provides a process flow diagram detailing the determination of an optimal detail level for image parcel presentation for a current viewing frustum in accordance with a preferred embodiment of the present invention.

DETAILED DESCRIPTION OF ONE OR MORE EMBODIMENTS

The preferred operational environment **10** of the present invention is generally shown in FIG. **1**. A network server system **12**, operating as a data store and server of image data, is responsive to requests received through a communications network, such as the Internet **14** generally and various tiers of internet service providers (ISPs) including a wireless connectivity provider **16**. Client systems, including conventional workstations and personal computers **18** and smaller, typically dedicated function devices often linked through wireless network connections, such as PDAs, webphones **20**, and automobile navigation systems, source image requests to the network server **12**, provide a client display and enable image navigational input by a user of the client system. Alternately, a dedicated function client system **20** may be connected through a separate or plug-in local network server **22**, preferably implementing a small, embedded Web server, to a fixed or removable storage local image repository **24**. Characteristically, the client system **18**, **20** displays are operated at some fixed resolution generally dependent on the underlying display hardware of the client systems **18**, **20**.

The image navigation capability supported by the present invention encompasses a viewing frustum placed within a three-dimensional space over the imaged displayed on the client **18**, **20**. Client user navigational inputs are supported to control the x, y lateral, rotational and z height positioning of the viewing frustum over the image as well as the camera angle of incidence relative to the plane of the image. To effect these controls, the software implemented on the client systems **18**, **20** supports a three-dimensional transform of the image data provided from the server **12**, **22**.

In accordance with the preferred embodiments of the present invention, as generally illustrated in FIG. **2**, a network image server system **30** stores a combination of source image data **32** and source overlay data **34**. The source image data **32** is typically high-resolution bit-map raster map and or satellite imagery of geographic regions, which can be obtained from commercial suppliers. The overlay image data **34** is typically a discrete data file providing image annotation information at defined coordinates relative to the source image data **32**. In the preferred embodiments of the present invention, image annotations include, for example, street, building and landmark names, as well as representative 2 and 3D objects, graphical icons, decals, line segments, and or text and or other characters, graphics and or other media.

The network image server system **30** preferably pre-processes the source image data **32** and or source overlay data **34** to forms preferred for storage and serving by the network server **12**, **22**. The source image data **32** is preferably pre-processed to obtain a series K.sub.1-N of derivative images of progressively lower image resolution. The source image data **32**, corresponding to the series image K.sub.0, is also subdivided into a regular array such that each resulting image parcel of the array has for example a 64 by 64 pixel resolution where the image data has a color or bit per pixel depth of 16 bits, which represents a data parcel size of 8K bytes. The resolution of the series K.sub.1-N of derivative images is preferably related to that of the source image data **32** or predecessor image in the series by a factor of four. The array subdivision is likewise related by a factor of four such that each image parcel is of a fixed 8K byte size.

In the preferred embodiment of the present invention, the image parcels are further compressed and stored by the network server **12**, **22**. The preferred compression algorithm may implements for example a fixed 4:1 compression ratio such that each compressed and stored image parcel has a fixed 2K byte size. The image parcels are preferably stored in a file of defined configuration such that any image parcel can be located by specification of a K.sub.D, X, Y value, representing the image set resolution index D and corresponding image array coordinate.

The source overlay data **34** is preferably pre-processed **36** into either an open XML format, such as the Geography Markup Language (GML), which is an XML based encoding standard for geographic information developed by the OpenGIS Consortium (OGC; www.opengis.org), or a proprietary binary representation. The XML/GML representation is preferred as permitting easier interchange between different commercial entities, while the binary representation is preferred as more compact and readily transferable to a client system **18**, **20**. In both cases, the source overlay data **34** is pre-processed to contain the annotation data preferably in a resolution independent form associated with a display coordinate specification relative to the source image data **32**. The XML, GML or binary overlay data may be compressed prior to storage on the network server **12**, **22**.

The preferred architecture **40** of a client system **18**, **20**, for purposes of implementing the present invention, is shown in FIG. **3**. The architecture **40** is preferably implemented by a software plug-in or application executed by the client system **18**, **20** and that utilizes basic software and hardware services provided by the client system **18**, **20**. A parcel request client **42** preferably implements an HTML client that supports HTML-based interactions with the server **12**, **22** using the underlying network protocol stack and hardware network interface provided by the client systems **18**, **20**. A central parcel processing control block **44** preferably implements the client process and control algorithms. The control block **44** directs the transfer of received image parcels and XML/GML/binary overlay data to a local parcel data store **46**. Preferably image data parcels are stored in conventional quad-tree data structures, where tree nodes of depth D correspond to the stored image parcels of a derivative image of resolution KD. The XML/GML/binary overlay data is preferably stored as a data object that can be subsequently read by an XML/GML/binary parser implemented as part of the control block **44**.

The control block **44** is also responsible for decompressing and directing the rendering of image parcels to a local display by a rendering engine **48**. Preferably, the rendering engine **48** writes to the video memory of the underlying client display hardware relying on only generic graphics acceleration hardware capabilities. In general, the relied on capabilities include bit-bit and related bit-oriented functions that are readily supported by current conventional display controller hardware. The rendering engine **48** is optimized to perform image parcel texture mapping without reliance on complex floating point operations, permitting even relatively simple processors to efficiently execute the rendering engine **48**.

Changes in the viewing frustum are determined from user input navigation commands by a frustum navigation block **50**. In the preferred embodiments of the present invention, the input navigation controls are modeled for three-dimensional fly-over navigation of the displayed image. The navigation controls support point-of-view rotation, translation, attitude, and altitude over the displayed image. The effective change in viewing frustum as determined by the frustum navigation block **50** is provided to the control block **44**.

The control block **44**, based in part on changes in the viewing frustum, determines the ordered priority of image parcels to be requested from the server **12**, **22** to support the progressive rendering of the displayed image. The image parcel requests are placed in a request queue **52** for issuance by the parcel request client **42**. Preferably, the pending requests are issued in priority order, thereby dynamically reflecting changes in the viewing frustum with minimum latency.

An optimal image parcel data flow **60**, as configured for use in the preferred embodiments of the present invention, is shown in FIG. **4**. Preferably, the TCP/IP network protocol is used to deliver image parcels to the clients **18**, **20**. For the preferred embodiments, where network bandwidth is limited or very limited, entire image parcels are preferably delivered in corresponding data packets. This preference maximizes data delivery while avoiding the substantial latency and processing overhead of managing image parcel data split over multiple network packets. Thus, a 2K byte compressed image parcel **62** is delivered as the data payload of a TCP/IP packet **64**. Uncompressed, the 8K byte image parcel **62** is recognized as part of the present invention as being within the nominally smallest L1 data cache **66** size of conventional microprocessors **68**. By ensuring that an uncompressed image parcel fits within the L1 cache, the texture map rendering algorithm can execute with minimum memory management overhead, thus optimally utilizing the processing capability of the microprocessor **68**. Additionally, the writing of video data as a product of the rendering algorithm is uniform, thereby improving the apparent video stability of the display to the user.

The client architecture **40** preferably executes in multiple process threads, with additional threads being utilized for individual network data request transactions. As shown in FIG. **5**, an image parcel management process **80** implements a loop that determines image parcels subject to update **82** and creates corresponding image parcel download requests **84**. Navigation events that alter the viewing frustum are considered in part to determine the current field of view. The quad-tree data structures are examined **86** to identify viewable image parcels of higher resolution than currently available in the parcel data store **46**.

A pool of image request threads is preferably utilized to manage the image parcel download operations. In the preferred embodiments of the present invention, a pool of four network request threads is utilized. The number of pool threads is determined as a balance between the available system resources and the network response latency, given the available bandwidth of the network connection. Empirically, for many wireless devices, four concurrent threads are able to support a relatively continuous delivery of image data parcels to the client **20** for display processing. As image parcels are progressively identified for download, a free request thread is employed to issue **88** a corresponding network request to the server **12**, **22**. When a network response is received, the corresponding thread recovers **90** the image parcel data. The received image parcel is then stored **92** in a corresponding quad-tree data structure node.

For small clients **20**, the available memory for the parcel data store **46** is generally quite restricted. In order to make optimal use of the available memory, only currently viewable image parcels are subject to download. Where the size of the parcel data store **46** is not so restricted, this constraint can be relaxed. In either case, a memory management process **94** runs to monitor use of the parcel data store **46** and selectively remove image parcels to free memory for newly requested image parcels. Preferably, the memory management process **94** operates to preferentially remove image parcels that are the furthest from the current viewing frustum and that have the highest data structure depth. Child node image parcels are always removed before a parent node parcel is removed.

A preferred network request management process **100** is shown in FIG. **6**. The process **100** waits **102** on the existence of a download request in the priority request queue **52**. The process **100** then waits on a network request pool thread to become free **104**. When a network request thread becomes available, the process **100** examines **106** all of the pending requests in the priority request queue **52** and selects **108** the request with the highest assigned priority. Thus, sequentially enqueued requests can be selectively issued out of order based on an independently assigned request priority. The request is then issued **110** and the request management process **100** leaves the request thread waiting on a network response.

FIG. **7** presents a preferred display management process **120**. Event driven user navigation information is evaluated **122** to determine a current viewing frustum location and orientation within a three-dimensional space relative to the displayed image. An algorithmic priority selection **124** of a next image parcel to render is then performed. The selected image parcel is then rendered **126** to the display memory **70**. The rendering operation preferably performs a texture map transform of the parcel data corresponding to the current viewing frustum location and orientation. The overlay data is then parsed or is pre-parsed to determine **128** whether the image coordinates of any overlay annotation correspond to the current image parcel location. If the coordinates match, the overlay annotation is rendered **130** to the video display memory **70**. The process **120** then continues with the next selection **124** of an image parcel to render, subject to any change in the viewing frustum location and orientation.

A preferred implementation of the selection **124** and rendering **126** of image parcels in accordance with the present invention is detailed in FIGS. **8** through **10**. Referring first to FIG. **8**, any outstanding requests in the priority request queue **52** are preferably cleared **142** in response to a change in the viewing frustum location and orientation. The effective altitude of the viewing frustum and or the resolution of the client display are then used as a basis for determining an optimal level of detail L that will be displayed. The detail level L value operates as a floor defining the maximum resolution K.sub.L of image data that can be effectively viewed on the client display given the location and or orientation of the viewing frustum. Constraining image parcel requests to the resolution range K.sub.N to K.sub.L, where K.sub.N is the lowest resolution derivative image stored by the network server **12**, **22**, prevents the download and processing of image parcels that cannot provide any perceptible improvement in the displayed image.

As part of the recursive evaluation of the optimal level of detail L, the image display space is progressively split 146 by four to one reductions into polygons. The quad-tree data structures holding existing image parcel data in the parcel data store 46 are concurrently traced 148 to establish a correspondence with the polygon map. Where the trace of a quad-tree data structure completes 150 to a node index of L for a polygon P, the node corresponding image parcel is associated with polygon P. The polygon P will not be further subdivided and no higher resolution image parcels will be requested for any portion of the image within the area represented by polygon P. Where the trace reaches a maximum node index of D for a polygon P' 152, where N.ltoreq.D<L and N is the index of the lowest resolution derivative image stored by the network server 12, 22, the image parcel associated with the node is associated with the polygon P'. This polygon P' will be subject to further subdivision and progressive requests for image parcels of higher resolution up to the detail level L.

Referring now to FIG. 9, a display image is then rendered 160 beginning with the maximum depth polygons previously found. Iterating over the set of maximum depth polygons, any polygons outside of the viewing frustum are skipped 162. Polygons that are at least partially visible are clipped to the applicable bounds of the viewing frustum 164. The polygon corresponding image parcel data is then texture mapped 166 into the polygon corresponding coordinates of the video memory 70. If the node index depth of the rendered image parcel is at least equal to the prior determined optimal detail level L 168, the iteration over the polygons P continues.

Where the node index depth is less than the optimal detail level L 170, the polygon P' is subdivided into four polygons and correspondingly represented by the creation of four child nodes within the associated quad-tree data structure 172. Four image parcel download requests are then created 174.

The download priority associated with each request is determined 176 by execution of a function S that operates on a 2D polygon argument P and returns a real number representing the request priority. The function argument P is a list of real (x, y) coordinates of the vertices of the current polygon in screen coordinates after being clipped to fit within the current viewing frustum. That is, the function S works over general polygons in a two-dimensional space, whose vertices are specified by the series {(x(1), y(1)), (x(2), y(2)), . . . , (x(n),y(n))}. The argument P vertices sent to S represent the position of the vertices composing each of the polygons, after being clipping to the viewing frustum, viewable within the display space having the fixed resolution [xRes, yRes]. Thus, the clipped polygons are all within the rectangle [0, xRes] x [0, yRes].

In execution of the function S, each of the P coordinates is first transformed by linear mapping of the screen coordinate space to the square [−1,1].times.[−1,1] by the operation x(i): =(x(i)−xRes/2)/(xRes/2); y(i)=(y(i)−yRes/2)/(yRes/2). The x and y coordinate values of each vertex (x(i),y(i)) for i=1 to n) are then transformed by the function T(a)=sgn(a)*pow(.vertline.a.vertline., d), where the control parameter d is a constant in the range (0,1], or equivalently the interval 0<d.ltoreq.1. The function S then returns a real value that is equal to the area covered by the argument polygon P vertices subject to the applied coordinate transformation. Thus, the accumulated priority for any image parcel pending download is the sum of the values of returned by the function S for each of the viewable polygons that require some part of the image parcel as the source data for texture map rendering of the polygon. The priority operation of the request queue 52 is such that download requests will be issued preferentially for image parcels with the largest priority value.

In accordance with the preferred embodiments of the present invention, the value of the control parameter d can be adjusted to ultimately affect the behavior of the function S in determining the download request priority. In general, image parcels with lower resolution levels will accumulate greater priority values due to the larger number of polygons that may use a given low resolution image parcel as a rendering data source. Such lower resolution image parcels are therefore more likely to be preferentially downloaded. In accordance with the present invention, this generally assures that a complete image of at least low resolution will be available for rendering.

The control parameter d, as applied in execution of the function S, well as the area distortion produced by the projection transform also influences the value returned by the function S such that relatively higher-resolution image parcels near the image view point will occasionally achieve a higher priority than relatively remote and partially viewed image parcels of lower resolution. Using values smaller than 1 for the control parameter d results in requests with a higher priority for parcels covering areas near the focal point of the viewer, which is presumed to be the center point of the display space, relative to requests for parcels further from the center point in absolute terms and of the same resolution depth D. Thus, in accordance with the present invention, the priority assigned to image parcel requests effectively influences the order of requests based on the relative contribution of the image parcel data to the total display quality of the image. Empirically, a value of 0.35 for the control parameter d for small screen devices, such as PDAs and webphones has been found to produce desirable results.

The computed priorities of each of the four newly created image parcel requests are then assigned 178 and the requests are enqueued in the priority request queue 52. The next polygon P is then considered in the loop of the image parcel rendering process 160.

The preferred algorithm 180 for determining the detail level L value for a given viewing frustum is shown in FIG. 10. In accordance with the present invention, the optimal detail level L is effectively the limit at which the resolution of image parcel data functionally exceeds the resolution of the client display. Preferably, to determine the optimal detail level L, the viewpoint or camera position of the viewing frustum is determined 182 relative to the displayed image. A nearest polygon P of depth D is then determined 184 from the effective altitude and attitude of the viewpoint. The nearest point A of the polygon P is then determined 186. The point A may be within the interior or an edge of the polygon P, though most likely be located at a vertex of the polygon P.

The optimum level of detail L at point A is then computed 188 as the base-4 logarithm of the number of pixels on the screen that would be covered by a single pixel from an image parcel of the lowest resolution K.sub.-N image, which is the quad-tree root image and corresponds to an image area covering the entire image map. The point A optimal detail level L is preferably computed analytically from the local value of the Jacobian of the projective transform used to transform the three dimensional image coordinate space to screen coordinates, evaluated at the point A.

Where the depth D of the polygon P is greater than the depth of the computed optimal level of detail L, the detail level L is taken as the optimal detail level L 190. Thus, through the process 140, an image parcel or corresponding section of the closest resolution image parcel associated with a parent node in the quad-tree data structure relative to the depth level L will be used as the texture for rendering the polygon P. Conversely, if the depth D is less than that of the optimal detail level L, the polygon P is effectively split into quadrants and the optimal level of detail is reevaluated. The process **180** thus continues iteratively until the optimal detail level L is found.

Thus, a system and methods of optimally presenting image data on client systems with potentially limited processing performance, resources, and communications bandwidth have been described. While the present invention has been described particularly with reference to the communications and display of geographic image data, the present invention is equally applicable to the efficient communications and display of other high resolution information.

In view of the above description of the preferred embodiments of the present invention, many modifications and variations of the disclosed embodiments will be readily appreciated by those of skill in the art. It is therefore to be understood that, within the scope of the appended claims, the invention may be practiced otherwise than as specifically described above.

The invention claimed is:

1. A method of retrieving large-scale images over network communications channels for display on a limited communication bandwidth computer device, said method comprising:

issuing, from a limited communication bandwidth computer device to a remote computer, a request for an update data parcel wherein the update data parcel is selected based on an operator controlled image viewpoint on the computer device relative to a predetermined image and the update data parcel contains data that is used to generate a display on the limited communication bandwidth computer device;

processing, on the remote computer, source image data to obtain a series K_{1-N} of derivative images of progressively lower image resolution and wherein series image K_0 being subdivided into a regular array wherein each resulting image parcel of the array has a predetermined pixel resolution wherein image data has a color or bit per pixel depth representing a data parcel size of a predetermined number of bytes, resolution of the series K_{1-N} of derivative images being related to that of the source image data or predecessor image in the series by a factor of two, and said array subdivision being related by a factor of two such that each image parcel being of a fixed byte size, wherein the processing further comprises compressing each data parcel and storing each data parcel on the remote computer in a file of defined configuration such that a data parcel can be located by specification of a K_D, X, Y value that represents the data set resolution index D and corresponding image array coordinate;

receiving said update data parcel from the data parcel stored in the remote computer over a communications channel; and

displaying on the limited communication bandwidth computer device using the update data parcel that is a part of said predetermined image, an image wherein said update data parcel uniquely forms a discrete portion of said predetermined image.

2. The method of claim **1**, wherein the update data parcel further comprises one of an image parcel textual mapping, a map parcel, a navigation cue, a text overlay and a topography.

3. The method of claim **1**, wherein the limited communication bandwidth computer device further comprises one of a mobile computer system, a cellular computer system, an embedded computer system, a handheld computer system, a personal digital assistants and an internet-capable digital phone.

4. The method of claim **1**, wherein the predetermined pixel resolution for each data parcel is a power of 2.

5. The method of claim **4**, wherein the predetermined pixel resolution is one of 32×32, 64×64, 128×128 and 256×256.

6. The method of claim **1** wherein said communications channel is a packetized communications channel and wherein said update data parcel is received from said packetized communications channel in one or more data packets.

7. The method of claim **6** wherein the data packet contains an update image parcel as a compressed data representation of said discrete portion of said predetermined image.

8. The method of claim **7** wherein said data packet contains said update image parcel as a fixed compression ratio representation of said discrete portion of said predetermined image.

9. The method of claim **7**, wherein said update image parcel contains pixel data in a fixed size array independent of the pixel resolution of said predetermined image.

10. The method of claim **1**, wherein issuing the request for an update data parcel further comprises preparing the request by associating a prioritization value to said request, wherein said prioritization value is based on the resolution of said update data parcel relative to that of other data parcels previously received by the limited communication bandwidth computer device, and wherein issuing said request is responsive to said prioritization value for issuing said request in a predefined prioritization order.

11. The method of claim **10**, wherein said prioritization values is based on the relative distance of said update data parcel from said operator controlled image viewpoint.

12. The method of claim **1**, wherein displaying the image further comprises multi-threading on the limited communication bandwidth computer device using the update data parcel to display the image.

13. A display system for displaying a large-scale image retrieved over a limited bandwidth communications channel, said display system comprising:

a display of defined screen resolution for displaying a defined image;

a memory providing for the storage of a plurality of image parcels displayable over respective portions of a mesh corresponding to said defined image;

a communications channel interface supporting the retrieval of a defined data parcel over a limited bandwidth communications channel;

a processor coupled between said display, memory and communications channel interface, said processor operative to select said defined data parcel, retrieve said defined data parcel via said limited bandwidth communications channel interface for storage in said memory, and render said defined data parcel over a discrete portion of said mesh to provide for a progressive resolution enhancement of said defined image on said display; and

a remote computer, coupled to the limited bandwidth communications channel, that delivers the defined data parcel wherein delivering the defined data parcel further comprises processing source image data to obtain a series K_{1-N} of derivative images of progressively lower image resolution and wherein series image K_0 being subdivided into a regular array wherein each resulting image parcel of the array has a predetermined pixel resolution wherein image data has a color or bit per pixel depth representing a data parcel size of a predetermined number of bytes, resolution of the series K_{1-N} of derivative images being related to that of the source image data or predecessor image in the series by a factor of two, and said array subdivision being related by a factor of two such that each image parcel being of a fixed byte size, wherein the processing further comprises compressing each data parcel and storing each data parcel on the remote computer in a file of defined configuration such that a data parcel can be located by specification of a K_D, X, Y value that represents the data set resolution index D and corresponding image array coordinate.

14. The display system of claim **13**, wherein said processor is responsive to said defined screen resolution and wherein said processor is operative to limit selection of said defined data parcel to where the resolution of said defined data parcel is less than or equal to said defined screen resolution.

15. The display system of claim **13**, wherein said processor is operative to prioritize the retrieval of said data parcel among a plurality of selected data parcels pending retrieval, wherein the relative priority of the data parcel is based on the difference in the resolution of the image parcel and the resolution of said plurality of selected data parcels.

16. The display system of claim **13**, wherein said processor is response to user navigation commands to define an image viewpoint relative to said defined image and wherein said processor is operative to prioritize the retrieval of said data parcel based on the distance between said image parcel and said image viewpoint relative to said defined image.

17. The display system of claim **13**, wherein the data parcel further comprises one of an image parcel textual mapping, a map parcel, a navigation cue, a text overlay and a topography.

18. The display system of claim **13**, wherein the predetermined pixel resolution for each data parcel is a power of 2.

19. The display system of claim **18**, wherein the predetermined pixel resolution is power of 2 and one of 32×32, 64×64, 128×128 and 256×256.

20. The display system of claim **13**, wherein the processor performs multi-threading to render said defined data parcel over the discrete portion of said mesh to provide for the progressive resolution enhancement of said defined image on said display.

* * * * *

EXHIBIT E

To the First Amended Complaint

US008924506B2

(12) **United States Patent**
Levanon et al.

(10) **Patent No.: US 8,924,506 B2**
(45) **Date of Patent: *Dec. 30, 2014**

(54) **OPTIMIZED IMAGE DELIVERY OVER LIMITED BANDWIDTH COMMUNICATION CHANNELS**

(75) Inventors: **Isaac Levanon**, Raanana, IL (US); **Yonatan Lavi**, Roanana, IL (US)

(73) Assignee: **Bradium Technologies LLC**, Suffern, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 865 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **13/027,929**

(22) Filed: **Feb. 15, 2011**

(65) **Prior Publication Data**

US 2011/0175914 A1 Jul. 21, 2011

Related U.S. Application Data

(63) Continuation-in-part of application No. 12/619,643, filed on Nov. 16, 2009, now Pat. No. 7,908,343, which is a continuation of application No. 10/035,987, filed on Dec. 24, 2001, now Pat. No. 7,644,131.

(60) Provisional application No. 60/258,488, filed on Dec. 27, 2000, provisional application No. 60/258,489, filed on Dec. 27, 2000, provisional application No. 60/258,465, filed on Dec. 27, 2000, provisional application No. 60/258,468, filed on Dec. 27, 2000, provisional application No. 60/258,466, filed on Dec. 27, 2000, provisional application No. 60/258,467, filed on Dec. 27, 2000.

(51) **Int. Cl.**

G06F 15/16	(2006.01)
G06F 3/14	(2006.01)
G06T 3/40	(2006.01)

(52) **U.S. Cl.**
CPC ***G06F 3/1454*** (2013.01); ***G06T 3/4092*** (2013.01); *G09G 2340/02* (2013.01); *G09G 2350/00* (2013.01)
USPC **709/217**; 709/203; 345/625; 382/232; 382/305

(58) **Field of Classification Search**
CPC G06T 3/4092; G09G 2340/02; G06F 3/14
USPC 709/202, 203, 217, 230, 246, 247; 345/625; 382/232, 305
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,929,860	A	7/1999	Hoppe
6,182,114	B1	1/2001	Yap et al.

(Continued)

OTHER PUBLICATIONS

Hoppe, H. "Progressive Meshes", SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, 1996, pp. 99-108.

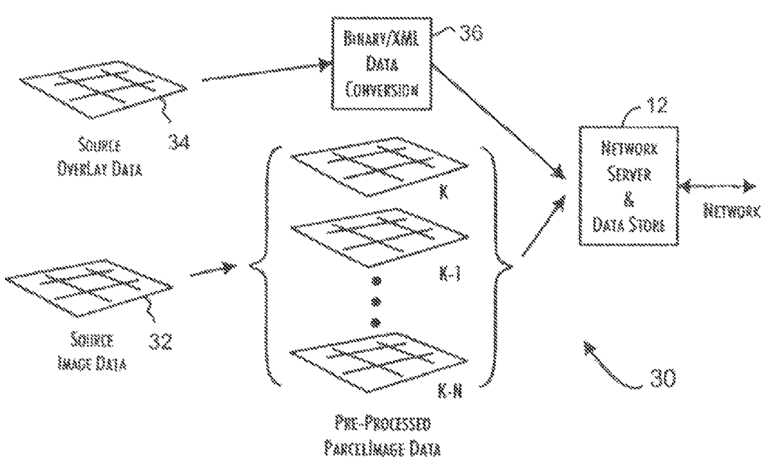
Primary Examiner — David Lazaro

(74) *Attorney, Agent, or Firm* — Anatoly S. Weiser, Esq.; Acuity Law Group, P.C.

(57) **ABSTRACT**

Large-scale images are retrieved over network communications channels for display on a client device by selecting an update image parcel relative to an operator controlled image viewpoint to display via the client device. A request is prepared for the update image parcel and associated with a request queue for subsequent issuance over a communications channel. The update image parcel is received from the communications channel and displayed as a discrete portion of the predetermined image. The update image parcel optimally has a fixed pixel array size, is received in a single and or plurality of network data packets, and were the fixed pixel array may be constrained to a resolution less than or equal to the resolution of the client device display.

21 Claims, 5 Drawing Sheets



(56) **References Cited**

U.S. PATENT DOCUMENTS

6,212,301	B1	4/2001	Warner et al.
6,314,452	B1	11/2001	Dekel et al.
6,326,965	B1	12/2001	Castelli et al.
6,345,279	B1	2/2002	Li et al.
6,346,938	B1	2/2002	Chan et al.
6,397,259	B1	5/2002	Lincke et al.
6,608,933	B1	8/2003	Dowell et al.
6,704,024	B2	3/2004	Robotham et al.
6,711,297	B1	3/2004	Chang et al.
6,801,665	B1	10/2004	Atsumi et al.

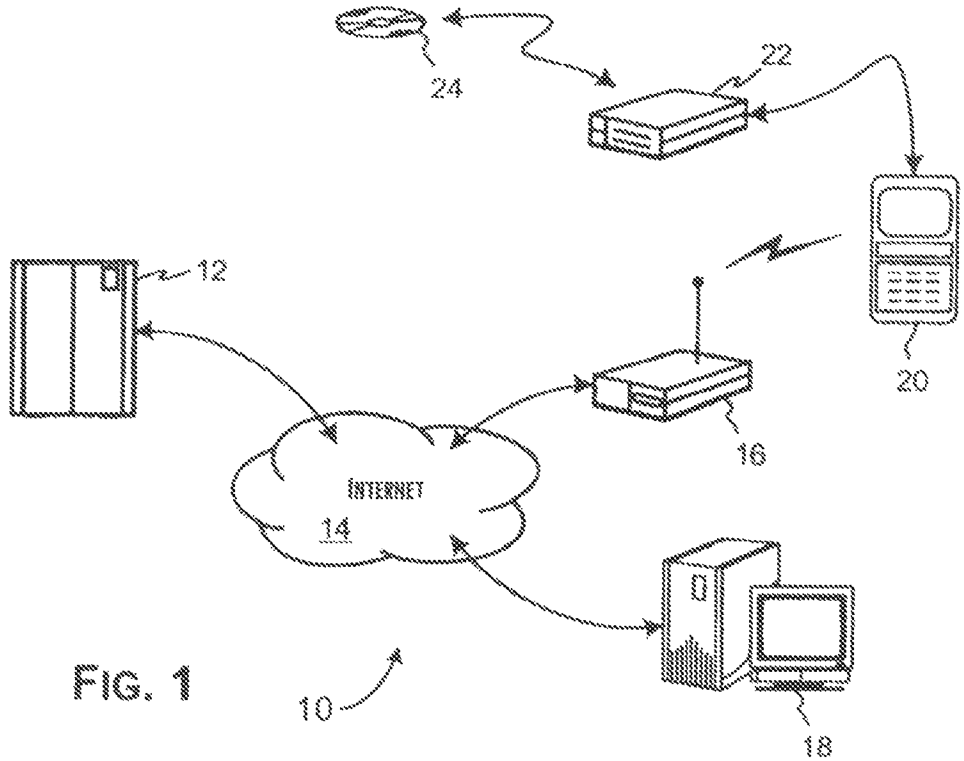

24
22
12
20
16
INTERNET
14
18
10

FIG. 1

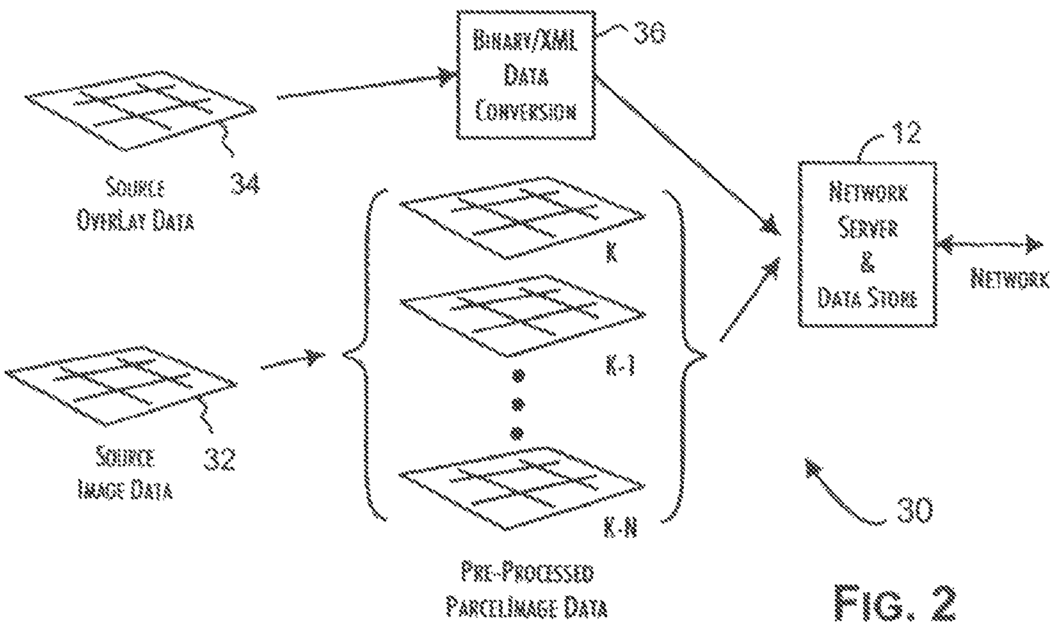

BINARY/XML
DATA
CONVERSION
36
SOURCE
OVERLAY DATA
34
12
NETWORK
SERVER
&
DATA STORE
NETWORK
K
K-1
K-N
SOURCE
IMAGE DATA
32
30
PRE-PROCESSED
PARCELIMAGE DATA

FIG. 2

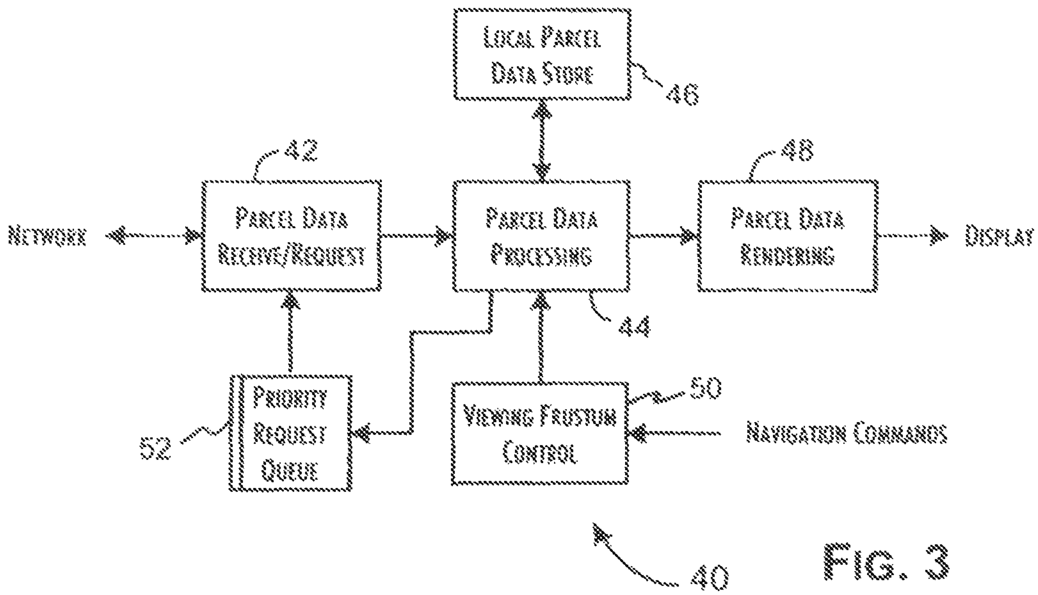

Local Parcel Data Store
46
42
48
Network
Parcel Data Receive/Request
Parcel Data Processing
Parcel Data Rendering
Display
44
50
52
Priority Request Queue
Viewing Frustum Control
Navigation Commands
40

FIG. 3

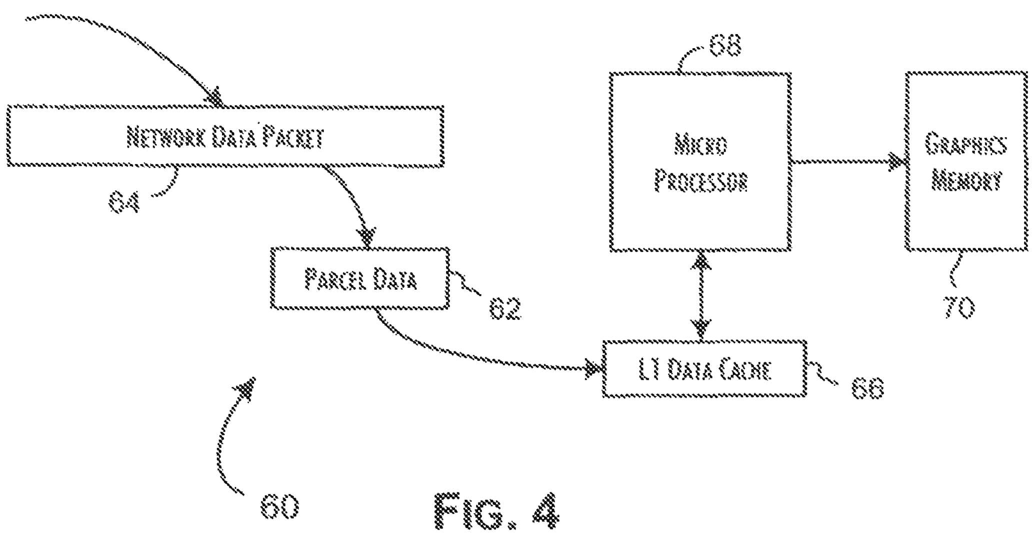
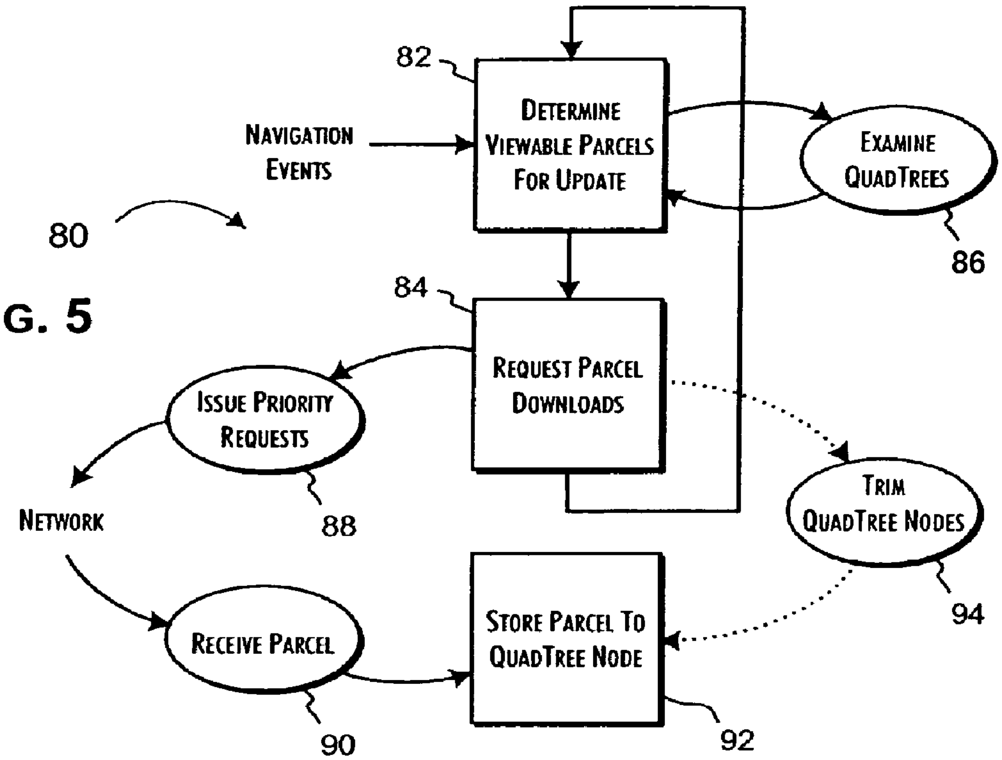
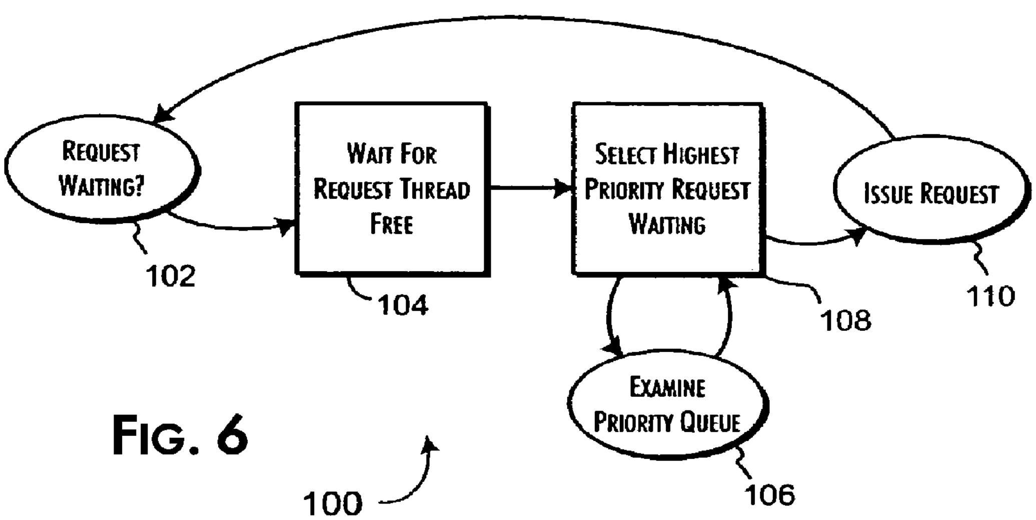

68
Network Data Packet
Micro Processor
Graphics Memory
64
Parcel Data
62
70
L1 Data Cache
66
60

FIG. 4



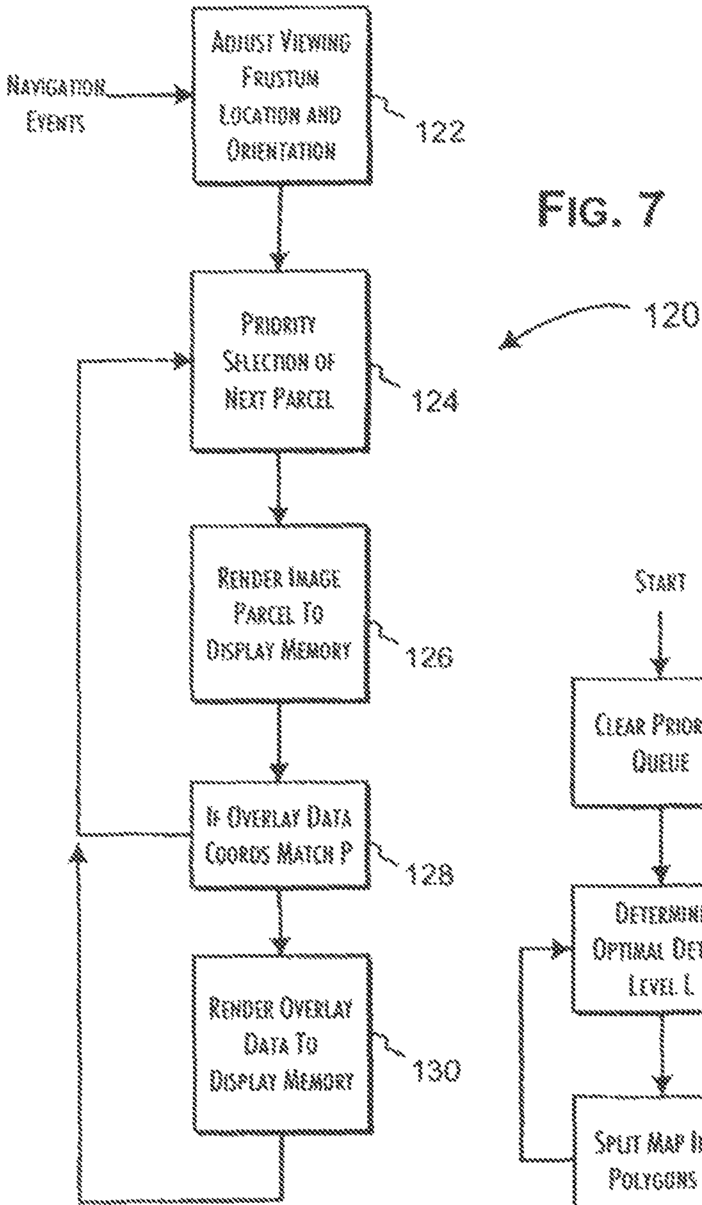
82
Navigation Events
Determine Viewable Parcels For Update
Examine QuadTrees
86
80
84
Request Parcel Downloads
Issue Priority Requests
88
Network
Trim QuadTree Nodes
94
Receive Parcel
Store Parcel To QuadTree Node
90
92

FIG. 5



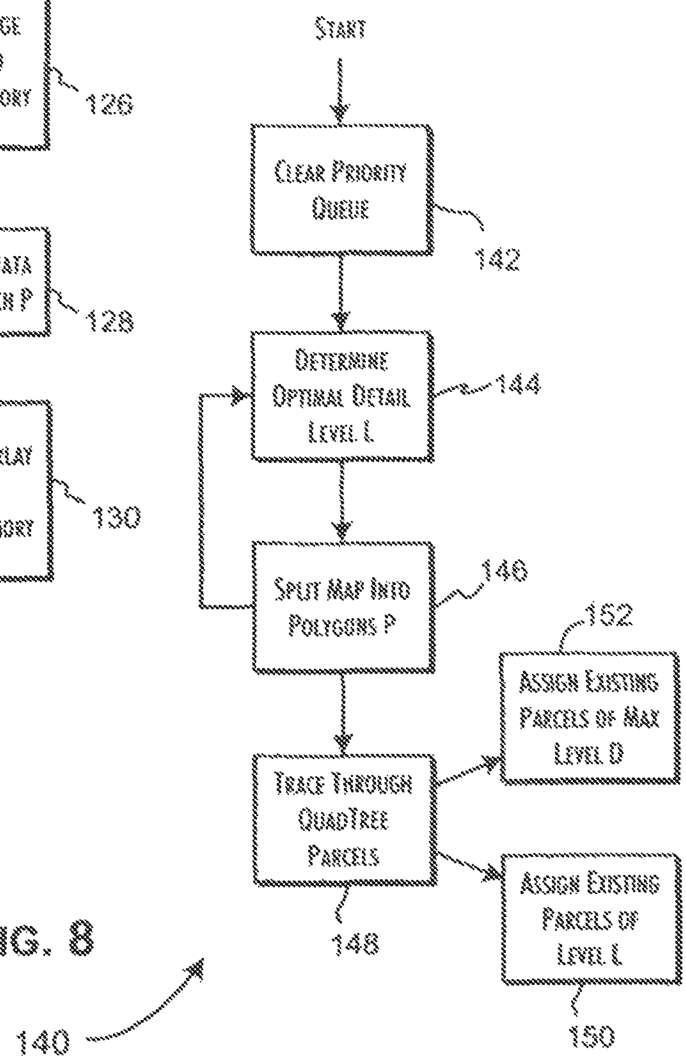
Request Waiting?
Wait For Request Thread Free
Select Highest Priority Request Waiting
Issue Request
102
104
108
110
Examine Priority Queue
106
100

FIG. 6



Navigation Events
Adjust Viewing Frustum Location and Orientation
122
Priority Selection of Next Parcel
124
Render Image Parcel To Display Memory
126
If Overlay Data Coords Match P
128
Render Overlay Data To Display Memory
130
120

FIG. 7



Start
Clear Priority Queue
142
Determine Optimal Detail Level L
144
Split Map Into Polygons P
146
Trace Through QuadTree Parcels
148
Assign Existing Parcels of Max Level D
152
Assign Existing Parcels of Level L
150
140

FIG. 8

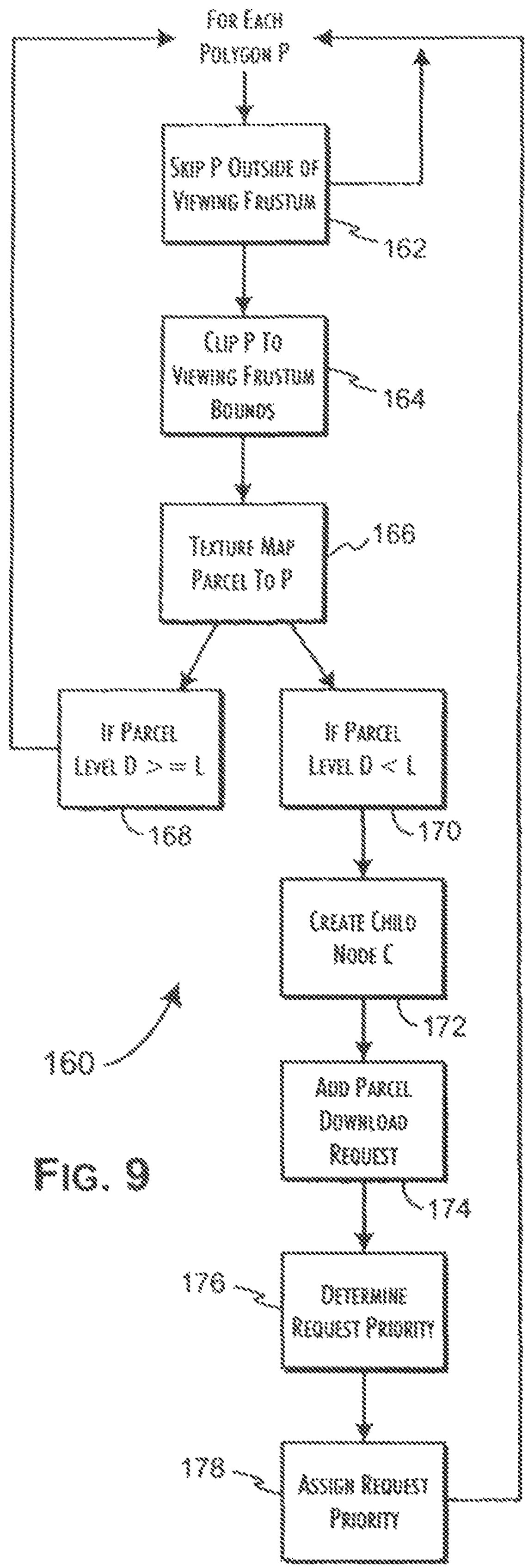

For Each Polygon P
Skip P Outside of Viewing Frustum
162
Clip P To Viewing Frustum Bounds
164
Texture Map Parcel To P
166
If Parcel Level D >= L
168
If Parcel Level D < L
170
Create Child Node C
172
Add Parcel Download Request
174
176
Determine Request Priority
178
Assign Request Priority
160

FIG. 9

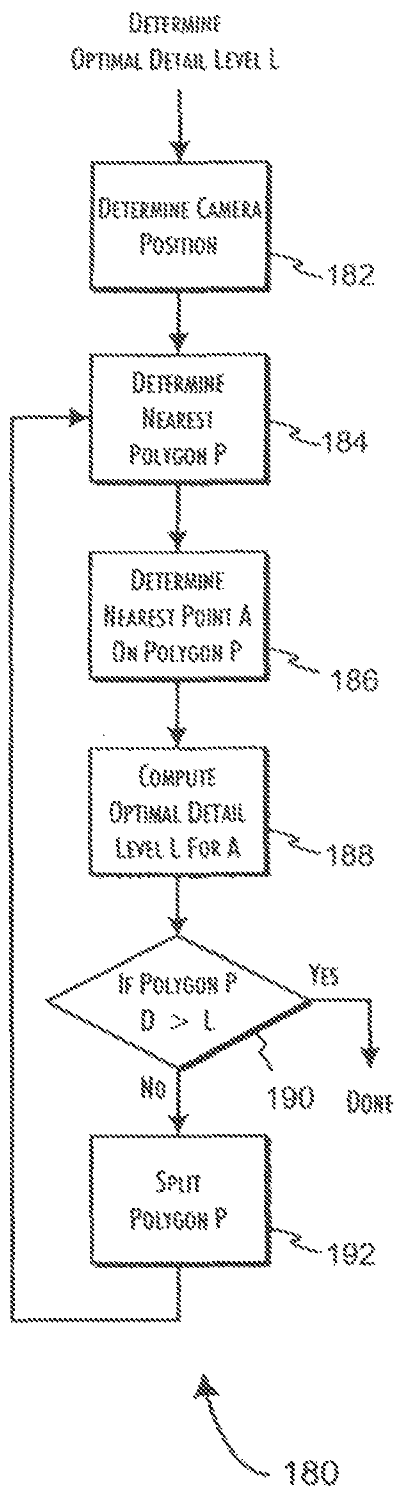

Determine Optimal Detail Level L
Determine Camera Position
182
Determine Nearest Polygon P
184
Determine Nearest Point A On Polygon P
186
Compute Optimal Detail Level L For A
188
If Polygon P D > L
Yes
No
190
Done
Split Polygon P
192
180

FIG. 10

OPTIMIZED IMAGE DELIVERY OVER LIMITED BANDWIDTH COMMUNICATION CHANNELS

PRIORITY CLAIMS/RELATED APPLICATIONS

This application is a continuation in part of and claims priority under 35 USC 120 to U.S. patent application Ser. No. 12/619,643 filed on Nov. 16, 2009 now U.S. Pat. No. 7,908, 343 which in turn in a continuation of and claims priority under 35 USC 120 to U.S. patent application Ser. No. 10/035, 987 filed on Dec. 24, 2001 and entitled "Optimized image delivery over limited bandwidth communication channels" (that now issued on Jan. 5, 2010 as U.S. Pat. No. 7,644,131) which in turn claims the benefit under 35 USC 119(*e*) of U.S. Provisional Application Nos. 60/258,488, 60/258,489, 60/258,465, 60/258,468, 60/258,466, and 60/258,467, all filed Dec. 27, 2000, all of which are incorporated herein by reference. The present application is also related to the co-pending application Ser. No. 10/035,981 entitled "System and Methods for Network Image Delivery with Dynamic Viewing Frustum Optimized for Limited Bandwidth Communication Channels, Levanon et al., filed on Dec. 24, 2001 (now U.S. Pat. No. 7,139,794 issued on Nov. 21, 2006 and which is assigned to the Assignee of the present Application.

FIELD

The disclosure is related to network based, image distribution systems and, in particular, to a system and methods for efficiently selecting and distributing image parcels through a narrowband or otherwise limited bandwidth communications channel to support presentation of high-resolution images subject to dynamic viewing frustums.

BACKGROUND

The Internet and or other network systems may provide a unique opportunity to transmit for example complex images, typically large scale bit-maps, particularly those approaching photo-realistic levels, over large area and or distances. In common application, the images may be geographic, topographic, and or other highly detailed maps. The data storage requirements and often proprietary nature of such images could be such that conventional interests may be to transfer the images on an as-needed basis.

In conventional fixed-site applications, the image data may be transferred over a relatively high-bandwidth network to client computer systems that in turn, may render the image. Client systems may typically implement a local image navigation system to provide zoom and or pan functions based on user interaction. As well recognized problem with such conventional systems could be that full resolution image presentation may be subject to the inherent transfer latency of the network. Different conventional systems have been proposed to reduce the latency affect by transmitting the image in highly compressed formats that support progressive resolution build-up of the image within the current client field of view. Using a transform compressed image transfer function increases the field of the image that can be transferred over a fixed bandwidth network in unit time. Progressive image resolution transmission, typically using a differential resolution method, permits an approximate image to be quickly presented with image details being continuously added over time.

Tzou, in U.S. Pat. No. 4,698,689, describes a two-dimensional data transform system that supports transmission of

differential coefficients to represent an image. Subsequent transmitted coefficient sets are progressively accumulated with prior transmitted sets to provide a succeedingly refined image. The inverse-transform function performed by the client computer is, however, highly compute intensive. In order to simplify the transform implementation and further reduce the latency of presenting any portion of an approximate image, images are sub-divided into a regular array. This enables the inverse-transform function on the client, which is time-critical, to deal with substantially smaller coefficient data sets. The array size in Tzou is fixed, which leads to progressively larger coefficient data sets as the detail level of the image increases. Consequently, there is an inherently increasing latency in resolving finer levels of detail.

An image visualization system proposed by Yap et al., U.S. Pat. No. 6,182,114, overcomes some of the foregoing problems. The Yap et al. system also employs a progressive encoding transform to compress the image transfer stream. The transform also operates on a subdivided image, but the division is indexed to the encoding level of the transform. The encoded transform coefficient data sets are, therefore, of constant size, which supports a modest improvement in the algorithmic performance of the inverse transform operation required on the client.

Yap et al. adds utilization of client image panning or other image pointing input information to support a foveation-based operator to influence the retrieval order of the subdivided image blocks. This two-dimensional navigation information is used to identify a foveal region that is presumed to be the gaze point of a client system user. The foveation operator defines the corresponding image block as the center point of an ordered retrieval of coefficient sets representing a variable resolution image. The gaze point image block represents the area of highest image resolution, with resolution reduction as a function of distance from the gaze point determined by the foveation operator. This technique thus progressively builds image resolution at the gaze point and succeedingly outward based on a relatively compute intensive function. Shifts in the gaze point can be responded to with relative speed by preferentially retrieving coefficient sets at and near the new foveal region.

Significant problems remain in permitting the convenient and effective use of complex images by many different types of client systems, even with the improvements provided by the various conventional systems. In particular, the implementation of conventional image visualization systems is generally unworkable for smaller, often dedicated or embedded, clients where use of image visualization would clearly be beneficial. Conventional approaches effectively presume that client systems have an excess of computing performance, memory and storage. Small clients, however, typically have restricted performance processors with possibly no dedicated floating-point support, little general purpose memory, and extremely limited persistent storage capabilities, particularly relative to common image sizes. A mobile computing device such as mobile phone, smart phone, tablet and or personal digital assistant (PDA) is a characteristic small client. Embedded, low-cost kiosk, automobile navigation systems and or Internet enabled/connected TV are other typical examples. Such systems are not readily capable, if at all, of performing complex, compute-intensive Fourier or wavelet transforms, particularly within a highly restricted memory address space.

As a consequence of the presumption that the client is a substantial computing system, conventional image visualization systems also presume that the client is supported by a complete operating system. Indeed, many expect and require an extensive set of graphics abstraction layers to be provided by the client system to support the presentation of the delivered image data. In general, these abstraction layers are conventionally considered required to handle the mapping of the image data resolution to the display resolution capabilities of the client system. That is, resolution resolved image data provided to the client is unconstrained by any limitation in the client system to actually display the corresponding image. Consequently, substantial processor performance and memory can be conventionally devoted to handling image data that is not or cannot be displayed.

Another problem is that small clients are generally constrained to generally to very limited network bandwidths, particularly when operating under wireless conditions. Such limited bandwidth conditions may exist due to either the direct technological constraints dictated by the use of a low bandwidth data channel or indirect constraints imposed on relatively high-bandwidth channels by high concurrent user loads. Cellular connected PDAs and webphones are examples of small clients that are frequently constrained by limited bandwidth conditions. The conventionally realizable maximum network transmission bandwidth for such small devices may range from below one kilobit per second to several tens of kilobits per second. While Yap et al. states that the described system can work over low bandwidth lines, little more than utilizing wavelet-based data compression is advanced as permitting effective operation at low communications bandwidths. While reducing the amount of data that must be carried from the server to the client is significant, Yap et al. simply relies on the data packet transfer protocols to provide for an efficient transfer of the compressed image data. Reliable transport protocols, however, merely mask packet losses and the resultant, sometimes extended, recovery latencies. When such covered errors occur, however, the aggregate bandwidth of the connection is reduced and the client system can stall waiting for further image data to process.

Consequently, there remains a need for an image visualization system that can support small client systems, place few requirements on the supporting client hardware and software resources, and efficiently utilize low to very low bandwidth network connections.

SUMMARY

Thus, a general purpose of the present invention is to provide an efficient system and methods of optimally presenting image data on client systems with potentially limited processing performance, resources, and communications bandwidth.

This is achieved in the present invention by providing for the retrieval of large-scale images over network communications channels for display on a client device by selecting an update image parcel relative to an operator controlled image viewpoint to display via the client device. A request is prepared for the update image parcel and associated with a request queue for subsequent issuance over a communications channel. The update image parcel is received from the communications channel and displayed as a discrete portion of the predetermined image. The update image parcel optimally has a fixed pixel array size, is received in a single and or plurality of network data packets, and were the fixed pixel array may be constrained to a resolution less than or equal to the resolution of the client device display.

An advantage of the present invention is that both image parcel data requests and the rendering of image data are optimized to address the display based on the display resolution of the client system.

Another advantage of the present invention is that the prioritization of image parcel requests is based on an adaptable parameter that minimizes the computational complexity of determining request prioritization and, in turn, the progressive improvement in display resolution within the field of view presented on a client display.

A further advantage of the present invention is that the client software system requires relatively minimal client processing power and storage capacity. Compute intensive numerical calculations are minimally required and image parcel data is compactly stored in efficient data structures. The client software system is very small and easily downloaded to conventional computer systems or embedded in conventional dedicated function devices, including portable devices, such as PDAs, tablets and webphones.

Still another advantage of the present invention is that image parcel data requests and presentation can be readily optimized to use low to very low bandwidth network connections. The software system of the present invention provides for re-prioritization of image parcel data requests and presentation in circumstances where the rate of point-of-view navigation exceeds the data request rate.

Yet another advantage of the present invention is that image parcel data rendering is performed without requiring any complex underlying hardware or software display subsystem. The client software system of the present invention includes a bit-map rendering engine that draws directly to the video memory of the display, thus placing minimal requirements on any underlying embedded or disk operating system and display drivers. Complex graphics and animation abstraction layers are not required.

Still another advantage of the present invention is that image parcel block compression is used to obtain fixed size transmission data blocks. Image parcel data is recoverable from transmission data using a relatively simple client decompression algorithm. Using fixed size transmission data blocks enables image data parcels to be delivered to the client in bounded time frames.

A yet further advantage of the present invention is that multiple data forms can be transferred to the client software system for concurrent display. Array overlay data, correlated positionally to the image parcel data and generally insensitive to image parcel resolution, can be initially or progressively provided to the client for parsing and parallel presentation on a client display image view.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other advantages and features of the present invention will become better understood upon consideration of the following detailed description of the invention when considered in connection with the accompanying drawings, in which like reference numerals designate like parts throughout the figures thereof, and wherein:

FIG. **1** depicts a preferred system environment within which various embodiments of the present invention can be utilized;

FIG. **2** is a block diagram illustrating the preparation of image parcel and overlay data set that are to be stored by and served from a network server system in accordance with a preferred embodiment of the present invention;

FIG. **3** is a block diagram of a client system image presentation system constructed in accordance with a preferred embodiment of the present invention;

FIG. **4** provides a data block diagram illustrating an optimized client image block processing path constructed in accordance with a preferred embodiment of the present invention;

FIG. **5** is a process flow diagram showing a main processing thread implemented in a preferred embodiment of the present invention;

FIG. **6** provides a process flow diagram showing a network request thread implemented in a preferred embodiment of the present invention;

FIG. **7** provides a process flow diagram showing a display image rendering thread implemented in a preferred embodiment of the present invention;

FIG. **8** provides a process flow diagram showing the parcel map processing performed preliminary to the rendering of image data parcels in accordance with a preferred embodiment of the present invention;

FIG. **9** provides a process flow diagram detailing the rendering and progressive prioritization of image parcel data download requests in accordance with a preferred embodiment of the present invention; and

FIG. **10** provides a process flow diagram detailing the determination of an optimal detail level for image parcel presentation for a current viewing frustum in accordance with a preferred embodiment of the present invention.

DETAILED DESCRIPTION OF ONE OR MORE EMBODIMENTS

The preferred operational environment **10** of the present invention is generally shown in FIG. **1**. A network server system **12**, operating as a data store and server of image data, is responsive to requests received through a communications network, such as the Internet **14** generally and various tiers of internet service providers (ISPs) including a wireless connectivity provider **16**. Client systems, including conventional workstations and personal computers **18** and smaller, typically dedicated function devices often linked through wireless network connections, such as PDAs, webphones **20**, and automobile navigation systems, source image requests to the network server **12**, provide a client display and enable image navigational input by a user of the client system. Alternately, a dedicated function client system **20** may be connected through a separate or plug-in local network server **22**, preferably implementing a small, embedded Web server, to a fixed or removable storage local image repository **24**. Characteristically, the client system **18**, **20** displays are operated at some fixed resolution generally dependent on the underlying display hardware of the client systems **18**, **20**.

The image navigation capability supported by the present invention encompasses a viewing frustum placed within a three-dimensional space over the imaged displayed on the client **18**, **20**. Client user navigational inputs are supported to control the x, y lateral, rotational and z height positioning of the viewing frustum over the image as well as the camera angle of incidence relative to the plane of the image. To effect these controls, the software implemented on the client systems **18**, **20** supports a three-dimensional transform of the image data provided from the server **12**, **22**.

In accordance with the preferred embodiments of the present invention, as generally illustrated in FIG. **2**, a network image server system **30** stores a combination of source image data **32** and source overlay data **34**. The source image data **32** is typically high-resolution bit-map raster map and or satellite imagery of geographic regions, which can be obtained from commercial suppliers. The overlay image data **34** is typically a discrete data file providing image annotation information at defined coordinates relative to the source image data **32**. In the preferred embodiments of the present invention, image annotations include, for example, street, building and landmark names, as well as representative 2 and 3D objects, graphical icons, decals, line segments, and or text and or other characters, graphics and or other media.

The network image server system **30** preferably pre-processes the source image data **32** and or source overlay data **34** to forms preferred for storage and serving by the network server **12**, **22**. The source image data **32** is preferably pre-processed to obtain a series K.sub.1-N of derivative images of progressively lower image resolution. The source image data **32**, corresponding to the series image K.sub.0, is also subdivided into a regular array such that each resulting image parcel of the array has for example a 64 by 64 pixel resolution where the image data has a color or bit per pixel depth of 16 bits, which represents a data parcel size of 8K bytes. The resolution of the series K.sub.1-N of derivative images is preferably related to that of the source image data **32** or predecessor image in the series by a factor of four. The array subdivision is likewise related by a factor of four such that each image parcel is of a fixed 8K byte size.

In the preferred embodiment of the present invention, the image parcels are further compressed and stored by the network server **12**, **22**. The preferred compression algorithm may implements for example a fixed 4:1 compression ratio such that each compressed and stored image parcel has a fixed 2K byte size. The image parcels are preferably stored in a file of defined configuration such that any image parcel can be located by specification of a K.sub.D, X, Y value, representing the image set resolution index D and corresponding image array coordinate.

In other implementations, the image array dimensions (which as 64×64 above) may be powers of two so that the image array can be used in texture mapping efficiently. To accommodate different data parcel size than the 2 KByte associated with 64×64 pixel parcel dimension described above and other communication protocol and overhead requirements, to accommodate transmission through other than a 3 KByte per second transmission channel, the present invention may use larger compression ratios that takes, for example, a 128×128 or 256×256 pixel parcel dimension and compresses it to meet the 3 KByte per second transmission channel, or other communication bandwidth used to stream the parcel.

The system may also accommodate different and larger data parcel sizes as transmission protocols, compression ratio achieved and micro-architectures of the client computers change. For purposes above, the data content was a pixel array representing image data. Where the data parcel content is vector, text or other data that may subject to different client system design factors, other parcel sizes may be used. Furthermore, the parcel sizes can be different between the server and the client. For example the server may create parcels or hold parcels, for streaming with 256×256 pixel parcel dimension and the client my render them as 64×64. In addition, parcels sizes on different servers may vary from one server to another and from the client side rendering. In the system, each grid is treated as a sparse data array that can be progressively revised to increase the resolution of the grid and thereby the level of detail presented by the grid.

The source overlay data **34** is preferably pre-processed **36** into either an open XML format, such as the Geography Markup Language (GML), which is an XML based encoding standard for geographic information developed by the OpenGIS Consortium (OGC; www.opengis.org), or a proprietary binary representation. The XML/GML representation is preferred as permitting easier interchange between different commercial entities, while the binary representation is preferred as more compact and readily transferable to a client system **18**, **20**. In both cases, the source overlay data **34** is pre-processed to contain the annotation data preferably in a resolution independent form associated with a display coordinate specification relative to the source image data **32**. The XML, GML or binary overlay data may be compressed prior to storage on the network server **12**, **22**.

The preferred architecture **40** of a client system **18**, **20**, for purposes of implementing the present invention, is shown in FIG. **3**. The architecture **40** is preferably implemented by a software plug-in or application executed by the client system **18**, **20** and that utilizes basic software and hardware services provided by the client system **18**, **20**. A parcel request client **42** preferably implements an HTML client that supports HTML-based interactions with the server **12**, **22** using the underlying network protocol stack and hardware network interface provided by the client systems **18**, **20**. A central parcel processing control block **44** preferably implements the client process and control algorithms. The control block **44** directs the transfer of received image parcels and XML/GML/binary overlay data to a local parcel data store **46**. Local parcel data store **46** may also act for example as local cache weather the entire data or part of it is in dynamic and/or static cache. Preferably image data parcels are stored in conventional quad-tree data structures, where tree nodes of depth D correspond to the stored image parcels of a derivative image of resolution KD. The XML/GML/binary overlay data is preferably stored as a data object that can be subsequently read by an XML/GML/binary parser implemented as part of the control block **44**.

The control block **44** is also responsible for decompressing and directing the rendering of image parcels to a local display by a rendering engine **48**. Preferably, the rendering engine **48** writes to the video memory of the underlying client display hardware relying on only generic graphics acceleration hardware capabilities and may take advantage of more advanced graphics acceleration hardware when available in the client system **18**, **20**. In general, the relied on capabilities include bit-bit and related bit-oriented functions that are readily supported by current conventional display controller hardware. The rendering engine **48** is optimized to perform image parcel texture mapping without reliance on complex floating point operations, permitting even relatively simple processors to efficiently execute the rendering engine **48**. The rendering engine **48** may take advantage of floating point operations when available in the client system **18**, **20**.

Changes in the viewing frustum are determined from user input navigation commands by a frustum navigation block **50**. In the preferred embodiments of the present invention, the input navigation controls are modeled for three-dimensional fly-over navigation of the displayed image. The navigation controls support point-of-view rotation, translation, attitude, and altitude over the displayed image. The effective change in viewing frustum as determined by the frustum navigation block **50** is provided to the control block **44**.

The control block **44**, based in part on changes in the viewing frustum, determines the ordered priority of image parcels to be requested from the server **12**, **22** to support the progressive rendering of the displayed image. The image parcel requests are placed in a request queue **52** for issuance by the parcel request client **42**. Preferably, the pending requests are issued in priority order, thereby dynamically reflecting changes in the viewing frustum with minimum latency.

In various implementations of the parcel processing, each data parcel is independently processable by the client system **18**,**20**, which is enabled by the selection and server-side processing used to prepare a parcel for transmission, thus providing for on-demand real-time parcel processing and creation on the server side for streaming based on the client request and not only for pre-processed parcel creation for retrieval for streaming from the server. Thus, the system can use both pre-processed parcels on the server and on-demand real-time creation of such parcels on the server side for streaming to the client.

An optimal image parcel data flow **60**, as configured for use in the preferred embodiments of the present invention, is shown in FIG. **4**. Preferably, the TCP/IP network protocol is used to deliver image parcels to the clients **18**, **20**. For the preferred embodiments, where network bandwidth is limited or very limited, entire image parcels are preferably delivered in corresponding data packets. This preference maximizes data delivery while avoiding the substantial latency and processing overhead of managing image parcel data split over multiple network packets. Thus, a 2K byte compressed image parcel **62** is delivered as the data payload of a TCP/IP packet **64**. Uncompressed, the 8K byte image parcel **62** is recognized as part of the present invention as being within the nominally smallest L1 data cache **66** size of conventional microprocessors **68**. By ensuring that an uncompressed image parcel fits within the L1 cache, the texture map rendering algorithm can execute with minimum memory management overhead, thus optimally utilizing the processing capability of the microprocessor **68**. Additionally, the writing of video data as a product of the rendering algorithm is uniform, thereby improving the apparent video stability of the display to the user.

The client architecture **40** preferably executes in multiple process threads, with additional threads being utilized for individual network data request transactions. As shown in FIG. **5**, an image parcel management process **80** implements a loop that determines image parcels subject to update **82** and creates corresponding image parcel download requests **84**. Navigation events that alter the viewing frustum are considered in part to determine the current field of view. The quad-tree data structures are examined **86** to identify viewable image parcels of higher resolution than currently available in the parcel data store **46**.

A pool of image request threads is preferably utilized to manage the image parcel download operations. In the preferred embodiments of the present invention, a pool of four network request threads is utilized. The number of pool threads is determined as a balance between the available system resources and the network response latency, given the available bandwidth of the network connection. Empirically, for many wireless devices, four concurrent threads are able to support a relatively continuous delivery of image data parcels to the client **20** for display processing. As image parcels are progressively identified for download, a free request thread is employed to issue 88 a corresponding network request to the server **12**, **22**. When a network response is received, the corresponding thread recovers **90** the image parcel data. The received image parcel is then stored **92** in a corresponding quad-tree data structure node.

For small clients **20**, the available memory for the parcel data store **46** is generally quite restricted. In order to make optimal use of the available memory, only currently viewable image parcels are subject to download. Where the size of the parcel data store **46** is not so restricted, this constraint can be relaxed. In either case, a memory management process **94** runs to monitor use of the parcel data store **46** and selectively remove image parcels to free memory for newly requested image parcels. Preferably, the memory management process **94** operates to preferentially remove image parcels that are the furthest from the current viewing frustum and that have the highest data structure depth. Preferably child node image parcels are always removed before a parent node parcel is removed.

A preferred network request management process **100** is shown in FIG. **6**. The process **100** waits **102** on the existence of a download request in the priority request queue **52**. The process **100** then waits on a network request pool thread to become free **104**. When a network request thread becomes available, the process **100** examines **106** all of the pending requests in the priority request queue **52** and selects **108** the request with the highest assigned priority. Thus, sequentially enqueued requests can be selectively issued out of order based on an independently assigned request priority. The request is then issued **110** and the request management process **100** leaves the request thread waiting on a network response.

FIG. **7** presents a preferred display management process **120**. Event driven user navigation information is evaluated **122** to determine a current viewing frustum location and orientation within a three-dimensional space relative to the displayed image. An algorithmic priority selection **124** of a next image parcel to render is then performed. The selected image parcel is then rendered **126** to the display memory **70**. The rendering operation preferably performs a texture map transform of the parcel data corresponding to the current viewing frustum location and orientation. The overlay data is then parsed or is pre-parsed to determine **128** whether the image coordinates of any overlay annotation correspond to the current image parcel location. If the coordinates match, the overlay annotation is rendered **130** to the video display memory **70**. The process **120** then continues with the next selection **124** of an image parcel to render, subject to any change in the viewing frustum location and orientation.

A preferred implementation of the selection **124** and rendering **126** of image parcels in accordance with the present invention is detailed in FIGS. **8** through **10**. Referring first to FIG. **8**, any outstanding requests in the priority request queue **52** are preferably cleared **142** in response to a change in the viewing frustum location and orientation. The effective altitude of the viewing frustum and or the resolution of the client display are then used as a basis for determining an optimal level of detail L that will be displayed. The detail level L value operates as a floor defining the maximum resolution K.sub.L of image data that can be effectively viewed on the client display given the location and or orientation of the viewing frustum. Constraining image parcel requests to the resolution range K.sub.N to K.sub.L, where K.sub.N is the lowest resolution derivative image stored by the network server **12**, **22**, prevents the download and processing of image parcels that cannot provide any perceptible improvement in the displayed image.

As part of the recursive evaluation of the optimal level of detail L, the image display space is progressively split **146** by four to one reductions into polygons. The quad-tree data structures holding existing image parcel data in the parcel data store **46** are concurrently traced **148** to establish a correspondence with the polygon map. Where the trace of a quad-tree data structure completes **150** to a node index of L for a polygon P, the node corresponding image parcel is associated with polygon P. The polygon P will not be further subdivided and no higher resolution image parcels will be requested for any portion of the image within the area represented by polygon P. Where the trace reaches a maximum node index of D for a polygon P' **152**, where N.ltoreq.D<L and N is the index of the lowest resolution derivative image stored by the network server **12**, **22**, the image parcel associated with the node is associated with the polygon P'. This polygon P' will be subject to further subdivision and progressive requests for image parcels of higher resolution up to the detail level L.

Referring now to FIG. **9**, a display image is then rendered **160** beginning with the maximum depth polygons previously found. Iterating over the set of maximum depth polygons, any polygons outside of the viewing frustum are skipped **162**. Polygons that are at least partially visible are clipped to the applicable bounds of the viewing frustum **164**. The polygon corresponding image parcel data is then texture mapped **166** into the polygon corresponding coordinates of the video memory **70**. If the node index depth of the rendered image parcel is at least equal to the prior determined optimal detail level L **168**, the iteration over the polygons P continues.

Where the node index depth is less than the optimal detail level L **170**, the polygon P' is subdivided into four polygons and correspondingly represented by the creation of four child nodes within the associated quad-tree data structure **172**. Four image parcel download requests are then created **174**.

The download priority associated with each request is determined **176** by execution of a function S that operates on a 2D polygon argument P and returns a real number representing the request priority. The function argument P is a list of real (x, y) coordinates of the vertices of the current polygon in screen coordinates after being clipped to fit within the current viewing frustum. That is, the function S works over general polygons in a two-dimensional space, whose vertices are specified by the series {(x(1),y(1)),(x(2),y(2)), . . . ,(x(n), y(n))}. The argument P vertices sent to S represent the position of the vertices composing each of the polygons, after being clipping to the viewing frustum, viewable within the display space having the fixed resolution [xRes, yRes]. Thus, the clipped polygons are all within the rectangle [0, xRes]×[0, yRes].

In execution of the function S, each of the P coordinates is first transformed by linear mapping of the screen coordinate space to the square [−1,1].times.[−1,1] by the operation x(i):=(x(i)−xRes/2)/(xRes/2); y(i)=(y(i)−yRes/2)/(yRes/2). The x and y coordinate values of each vertex (x(i),y(i)) for i=1 to n) are then transformed by the function T(a)=sgn(a)*pow (.vertline.a.vertline., d), where the control parameter d is a constant in the range (0,1], or equivalently the interval 0<dltoreq.1. The function S then returns a real value that is equal to the area covered by the argument polygon P vertices subject to the applied coordinate transformation. Thus, the accumulated priority for any image parcel pending download is the sum of the values of returned by the function S for each of the viewable polygons that require some part of the image parcel as the source data for texture map rendering of the polygon. The priority operation of the request queue **52** is such that download requests will be issued preferentially for image parcels with the largest priority value.

In accordance with the preferred embodiments of the present invention, the value of the control parameter d can be adjusted to ultimately affect the behavior of the function S in determining the download request priority. In general, image parcels with lower resolution levels will accumulate greater priority values due to the larger number of polygons that may use a given low resolution image parcel as a rendering data source. Such lower resolution image parcels are therefore more likely to be preferentially downloaded. In accordance with the present invention, this generally assures that a complete image of at least low resolution will be available for rendering.

The control parameter d, as applied in execution of the function S, well as the area distortion produced by the projection transform also influences the value returned by the function S such that relatively higher-resolution image parcels near the image view point will occasionally achieve a higher priority than relatively remote and partially viewed image parcels of lower resolution. Using values smaller than 1 for the control parameter d results in requests with a higher priority for parcels covering areas near the focal point of the viewer, which is presumed to be the center point of the display space, relative to requests for parcels further from the center point in absolute terms and of the same resolution depth D. Thus, in accordance with the present invention, the priority assigned to image parcel requests effectively influences the order of requests based on the relative contribution of the image parcel data to the total display quality of the image. Empirically, a value of 0.35 for the control parameter d for small screen devices, such as PDAs and webphones has been found to produce desirable results.

The computed priorities of each of the four newly created image parcel requests are then assigned **178** and the requests are enqueued in the priority request queue **52**. The next polygon P is then considered in the loop of the image parcel rendering process **160**.

The preferred algorithm **180** for determining the detail level L value for a given viewing frustum is shown in FIG. **10**. In accordance with the present invention, the optimal detail level L is effectively the limit at which the resolution of image parcel data functionally exceeds the resolution of the client display. Preferably, to determine the optimal detail level L, the viewpoint or camera position of the viewing frustum is determined **182** relative to the displayed image. A nearest polygon P of depth D is then determined **184** from the effective altitude and attitude of the viewpoint. The nearest point A of the polygon P is then determined **186**. The point A may be within the interior or an edge of the polygon P, though most likely be located at a vertex of the polygon P.

The optimum level of detail L at point A is then computed **188** as the base-4 logarithm of the number of pixels on the screen that would be covered by a single pixel from an image parcel of the lowest resolution K.sub.-N image, which is the quad-tree root image and corresponds to an image area covering the entire image map. The point A optimal detail level L is preferably computed analytically from the local value of the Jacobian of the projective transform used to transform the three dimensional image coordinate space to screen coordinates, evaluated at the point A.

Where the depth D of the polygon P is greater than the depth of the computed optimal level of detail L, the detail level L is taken as the optimal detail level L **190**. Thus, through the process **140**, an image parcel or corresponding section of the closest resolution image parcel associated with a parent node in the quad-tree data structure relative to the depth level L will be used as the texture for rendering the polygon P. Conversely, if the depth D is less than that of the optimal detail level L, the polygon P is effectively split into quadrants and the optimal level of detail is reevaluated. The process **180** thus continues iteratively until the optimal detail level L is found.

Thus, a system and methods of optimally presenting image data on client systems with potentially limited processing performance, resources, and communications bandwidth have been described. While the present invention has been described particularly with reference to the communications and display of geographic image data, the present invention is equally applicable to the efficient communications and display of other high resolution information.

In the process implemented by the system described above, data parcels may be selected for sequential transmission based on a prioritization of the importance of the data contained. The criteria for the importance of a particular data parcel may be defined as suitable for particular applications and may directly relate to the presentation of image quality, provision of a textual overlay of a low-quality image to quickly provide a navigational orientation, or the addition of topography information at a rate or timing different from the rate of image quality improvement. Thus, image data layers reflecting navigational cues, text overlays, and topography can be composed into data packets for transmission subject to prioritizations set by the server alone and not based on the client system and interactively influenced by the actions and commands provided by the user of the client system. However this also may be influenced based on the nature and type of the client system, and interactively influenced by the actions and commands provided by the user of the client system (FIG. **5**).

In view of the above description of the preferred embodiments of the present invention, many modifications and variations of the disclosed embodiments will be readily appreciated by those of skill in the art. It is therefore to be understood that, within the scope of the appended claims, the invention may be practiced otherwise than as specifically described above.

The invention claimed is:

1. A method of retrieving large-scale images over network communications channels for display on a limited communication bandwidth computer device, said method comprising:

issuing, from a limited communication bandwidth computer device to a remote computer, a request for an update data parcel wherein the update data parcel is selected based on an operator controlled image viewpoint on the computer device relative to a predetermined image and the update data parcel contains data that is used to generate a display on the limited communication bandwidth computer device;

processing, on the remote computer, source image data to obtain a series K_{1-N} of derivative images of progressively lower image resolution and wherein series image K_0 being subdivided into a regular array wherein each resulting image parcel of the array has a predetermined pixel resolution wherein image data has a color or bit per pixel depth representing a data parcel size of a predetermined number of bytes, resolution of the series K_{1-N} of derivative images being related to that of the source image data or predecessor image in the series by a factor of two, and said array subdivision being related by a factor of two such that each image parcel being of a fixed byte size;

receiving said update data parcel from the data parcel stored in the remote computer over a communications channel; and

displaying on the limited communication bandwidth computer device using the update data parcel that is a part of said predetermined image, an image wherein said update data parcel uniquely forms a discrete portion of said predetermined image.

2. The method of claim **1**, wherein processing the source image data further comprises one of pre-processing the source image data on the remote computer and processing the source image data in real-time on-demand based on the request for the updated image parcel.

3. The method of claim **2**, wherein receiving the update data parcel over a communications channel further comprises streaming the update data parcel over a communications channel to the limited communication bandwidth computer device.

4. The method of claim **1**, wherein the limited communication bandwidth computer device further comprises one of a mobile computer system, a cellular computer system, an embedded computer system, a handheld computer system, a personal digital assistants and an internet-capable digital phone and a television.

5. The method of claim **1**, wherein a size of the data parcel on the remote computer is different from the update data parcel on the limited communication bandwidth computer device.

6. The method of claim **1**, wherein processing the source image data further comprises queuing the update data parcels on the remote computer based on an importance of the update data parcel as determined by the remote computer.

7. The method of claim **1**, wherein the processing further comprises compressing each data parcel and storing each data parcel on the remote computer in a file of defined configuration such that a data parcel can be located by specification of a K_D, X, Y value that represents the data set resolution index D and corresponding image array coordinate.

8. A display system for displaying a large-scale image retrieved over a limited bandwidth communications channel, said display system comprising:

a display of defined screen resolution for displaying a defined image;

a memory providing for the storage of a plurality of image parcels displayable over respective portions of a mesh corresponding to said defined image;

a communications channel interface supporting the retrieval of a defined data parcel over a limited bandwidth communications channel;

a processor coupled between said display, memory and communications channel interface, said processor operative to select said defined data parcel, retrieve said defined data parcel via said limited bandwidth communications channel interface for storage in said memory, and render said defined data parcel over a discrete portion of said mesh to provide for a progressive resolution enhancement of said defined image on said display; and

wherein a remote computer coupled to the limited bandwidth communications channel, delivers the defined data parcel wherein delivering the defined data parcel further comprises processing source image data to obtain a series K_{1-N} of derivative images of progressively lower image resolution and wherein series image K_0 being subdivided into a regular array wherein each resulting image parcel of the array has a predetermined pixel resolution wherein image data has a color or bit per pixel depth representing a data parcel size of a predetermined number of bytes, resolution of the series K_{1-N} of derivative images being related to that of the source image data or predecessor image in the series by a factor of two, and said array subdivision being related by a factor of two such that each image parcel being of a fixed byte size.

9. The display system of claim **8**, wherein processing the source image data further comprises one of pre-processing the source image data on the remote computer and processing the source image data in real-time on-demand based on the request for the updated image parcel.

10. The display system of claim **9**, wherein receiving the update data parcel over a communications channel further comprises streaming the update data parcel over a communications channel to the limited communication bandwidth computer device.

11. The display system of claim **8**, wherein the limited communication bandwidth computer device further comprises one of a mobile computer system, a cellular computer system, an embedded computer system, a handheld computer system, a personal digital assistants and an internet-capable digital phone and a television.

12. The display system of claim **8**, wherein a size of the data parcel on the remote computer is different from the update data parcel on the limited communication bandwidth computer device.

13. The display system of claim **8**, wherein processing the source image data further comprises queuing the update data parcels on the remote computer based on an importance of the update data parcel as determined by the remote computer.

14. The display system of claim **8**, wherein the processing may further comprises compressing each data parcel and storing each data parcel on the remote computer in a file of defined configuration such that a data parcel can be located by specification of a K_D, X, Y value that represents the data set resolution index D and corresponding image array coordinate.

15. A remote computer for delivering large-scale images over network communications channels for display on a limited communication bandwidth computer device that has a display system for displaying a large-scale image retrieved over a limited bandwidth communications channel, a display of defined screen resolution for displaying a defined image, a memory providing for the storage of a plurality of image parcels displayable over respective portions of a mesh corresponding to said defined image, a communications channel interface supporting the retrieval of a defined data parcel over a limited bandwidth communications channel and a processor coupled between said display, memory and communications channel interface, said processor operative to select said defined data parcel, retrieve said defined data parcel via said limited bandwidth communications channel interface for storage in said memory, and render said defined data parcel over a discrete portion of said mesh to provide for a progressive resolution enhancement of said defined image on said display, the remote computer comprises:

a parcel processing unit that processes a piece of source image data and delivers the defined data parcel to the limited communication bandwidth computer device; and

wherein the parcel processing unit further comprises a parcel processing control that processes source image data to obtain a series K_{1-N} of derivative images of progressively lower image resolution and wherein series image K_0 being subdivided into a regular array wherein each resulting image parcel of the array has a predetermined pixel resolution wherein image data has a color or bit per pixel depth representing a data parcel size of a predetermined number of bytes, resolution of the series K_{1-N} of derivative images being related to that of the source image data or predecessor image in the series by a factor of two, and said array subdivision being related by a factor of two such that each image parcel being of a fixed byte size.

16. The remote computer of claim **15**, wherein processing the source image data further comprises one of pre-processing the source image data on the remote computer and processing the source image data in real-time on-demand based on the request for the updated image parcel.

17. The remote computer of claim **16**, wherein receiving the update data parcel over a communications channel further comprises streaming the update data parcel over a communications channel to the limited communication bandwidth computer device.

18. The remote computer of claim **15**, wherein the limited communication bandwidth computer device further comprises one of a mobile computer system, a cellular computer system, an embedded computer system, a handheld computer system, a personal digital assistants and an internet-capable digital phone and a television.

19. The remote computer of claim **15**, wherein a size of the data parcel on the remote computer is different from the update data parcel on the limited communication bandwidth computer device.

20. The remote computer of claim **15**, wherein processing the source image data further comprises queuing the update data parcels on the remote computer based on an importance of the update data parcel as determined by the remote computer.

21. The remote computer of claim **15**, wherein processing further comprises compressing each data parcel and storing each data parcel on the remote computer in a file of defined configuration such that a data parcel can be located by specification of a K_D, X, Y value that represents the data set resolution index D and corresponding image array coordinate.

* * * * *

EXHIBIT F

To the First Amended Complaint

US 20110175914A1

(19) **United States**
(12) **Patent Application Publication**
Levanon et al.
(10) **Pub. No.: US 2011/0175914 A1**
(43) **Pub. Date: Jul. 21, 2011**

(54) **OPTIMIZED IMAGE DELIVERY OVER LIMITED BANDWIDTH COMMUNICATION CHANNELS**

(75) Inventors: **Isaac Levanon**, Raanana, IL (US); **Yonatan Lavi**, Roanana (IL)

(73) Assignee: **INOVO LIMITED**

(21) Appl. No.: **13/027,929**

(22) Filed: **Feb. 15, 2011**

Related U.S. Application Data

(63) Continuation-in-part of application No. 12/619,643, filed on Nov. 16, 2009, now Pat. No. 7,908,343, which is a continuation of application No. 10/035,987, filed on Dec. 24, 2001, now Pat. No. 7,644,131.

(60) Provisional application No. 60/258,488, filed on Dec. 27, 2000, provisional application No. 60/258,489, filed on Dec. 27, 2000, provisional application No. 60/258,465, filed on Dec. 27, 2000, provisional application No. 60/258,468, filed on Dec. 27, 2000, provisional application No. 60/258,466, filed on Dec. 27, 2000, provisional application No. 60/258,467, filed on Dec. 27, 2000.

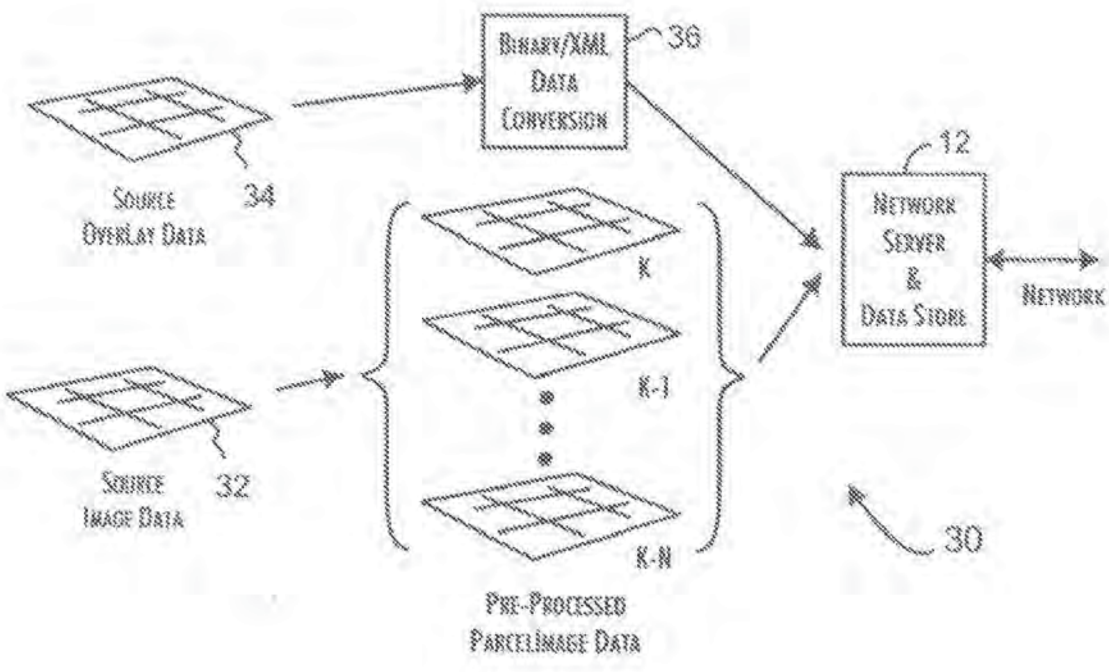
Publication Classification

(51) **Int. Cl.**
G06F 15/16 (2006.01)
G06K 9/36 (2006.01)
G06T 17/00 (2006.01)

(52) **U.S. Cl.** **345/428**; 709/217; 382/232; 709/231

(57) **ABSTRACT**

Large-scale images are retrieved over network communications channels for display on a client device by selecting an update image parcel relative to an operator controlled image viewpoint to display via the client device. A request is prepared for the update image parcel and associated with a request queue for subsequent issuance over a communications channel. The update image parcel is received from the communications channel and displayed as a discrete portion of the predetermined image. The update image parcel optimally has a fixed pixel array size, is received in a single and or plurality of network data packets, and were the fixed pixel array may be constrained to a resolution less than or equal to the resolution of the client device display.



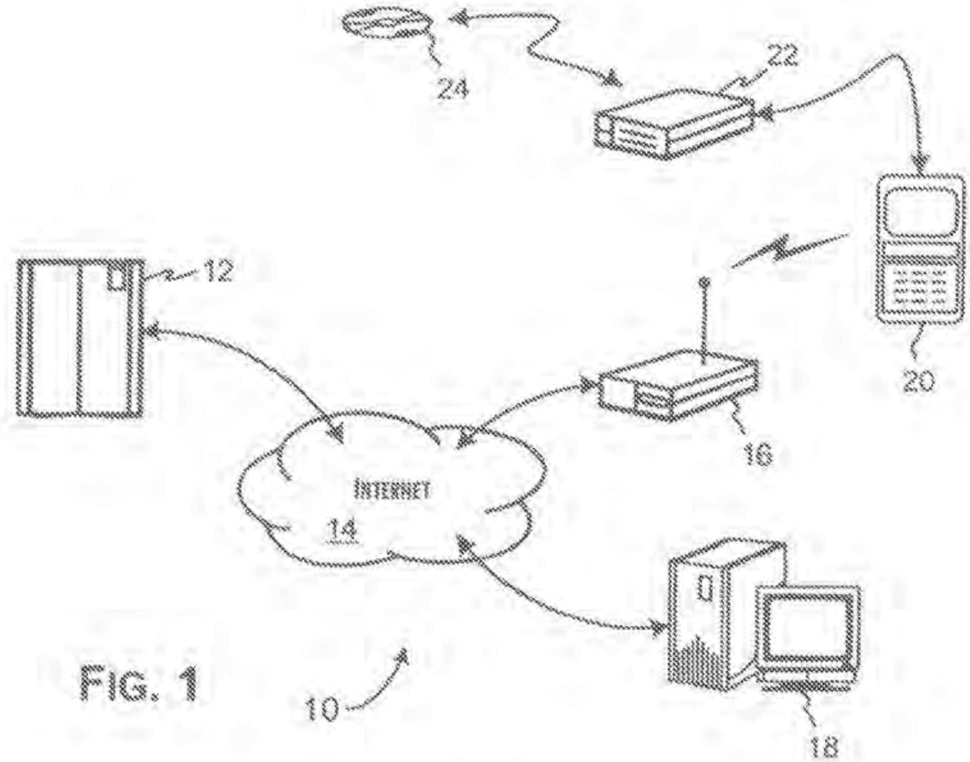

24
22
12
20
16
INTERNET
14
10
18

FIG. 1

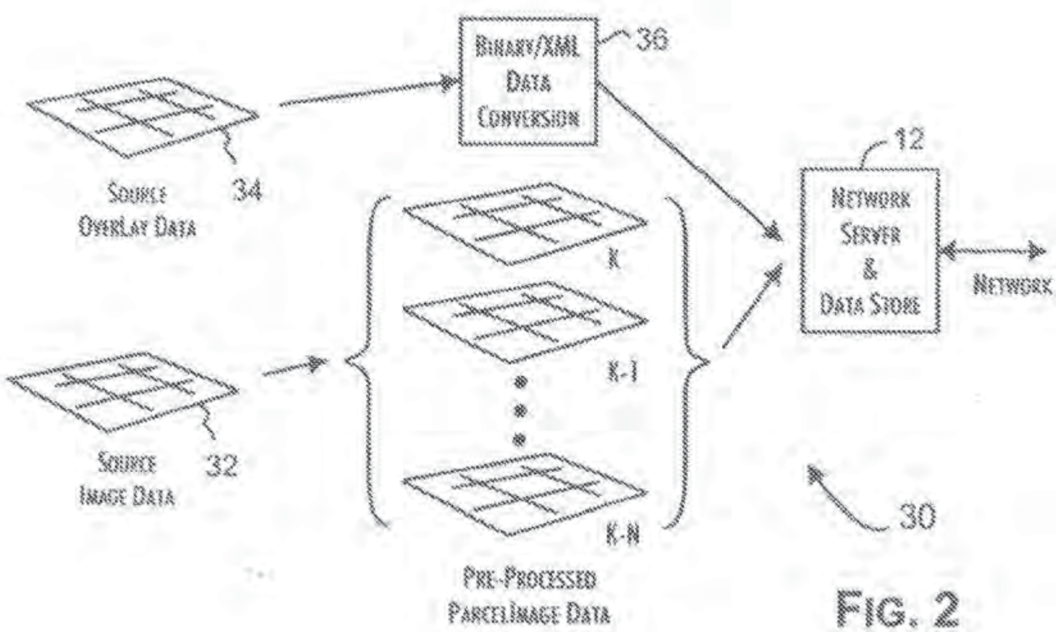
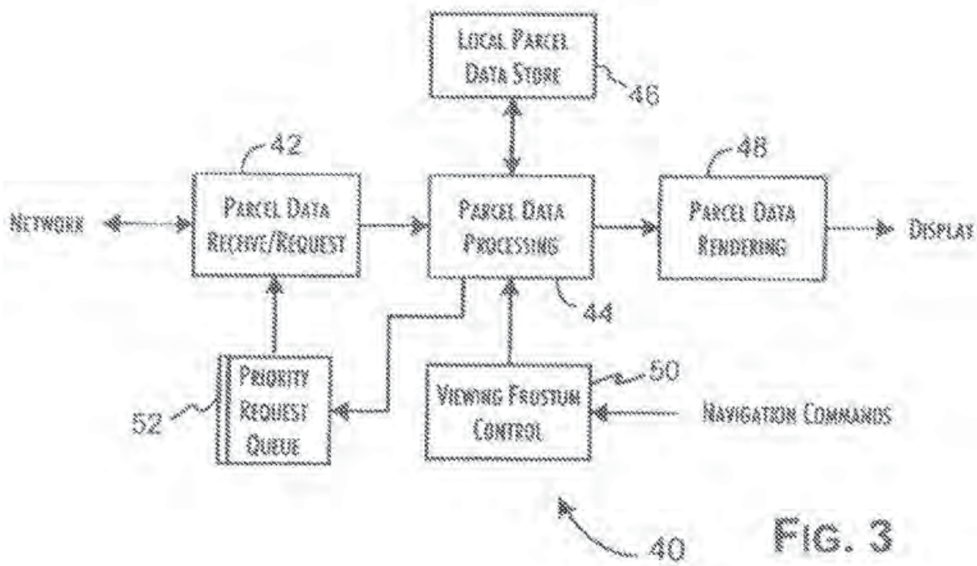
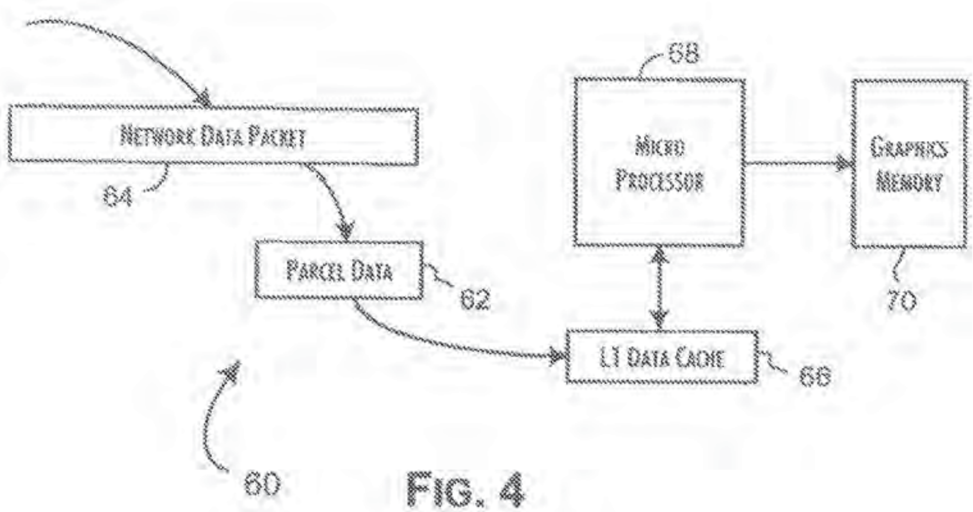

BINARY/XML
DATA
CONVERSION
36
12
SOURCE
OVERLAY DATA
34
NETWORK
SERVER
&
DATA STORE
NETWORK
K
K-1
SOURCE
IMAGE DATA
32
K-N
30
PRE-PROCESSED
PARCELIMAGE DATA

FIG. 2



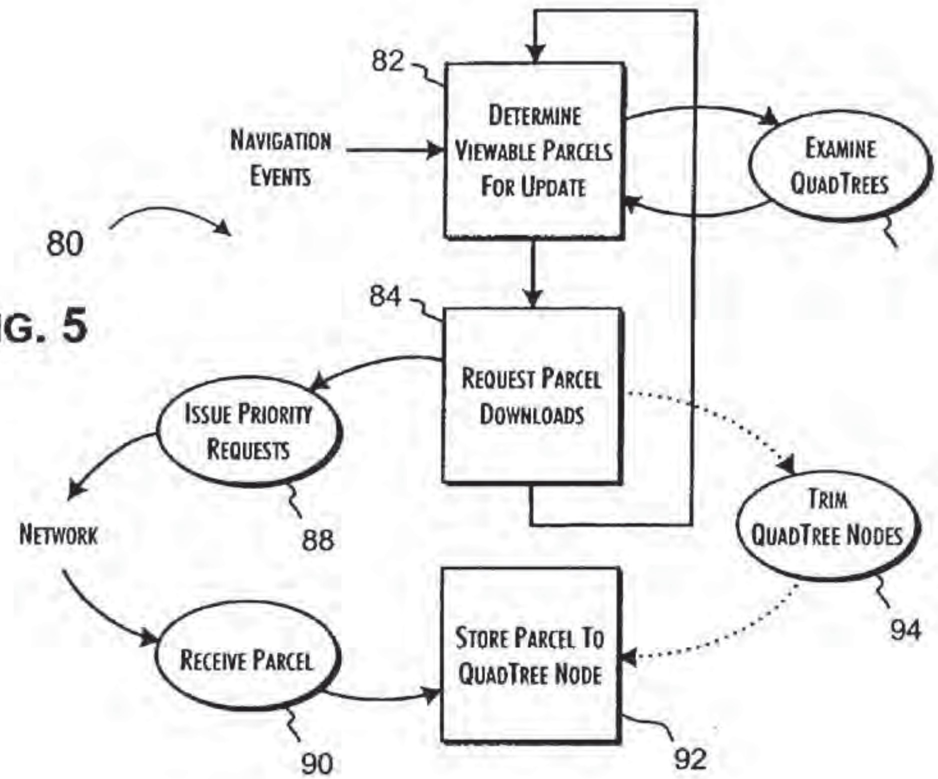
LOCAL PARCEL DATA STORE
46
42
48
NETWORK
PARCEL DATA RECEIVE/REQUEST
PARCEL DATA PROCESSING
PARCEL DATA RENDERING
DISPLAY
44
50
52
PRIORITY REQUEST QUEUE
VIEWING FRUSTUM CONTROL
NAVIGATION COMMANDS
40

FIG. 3



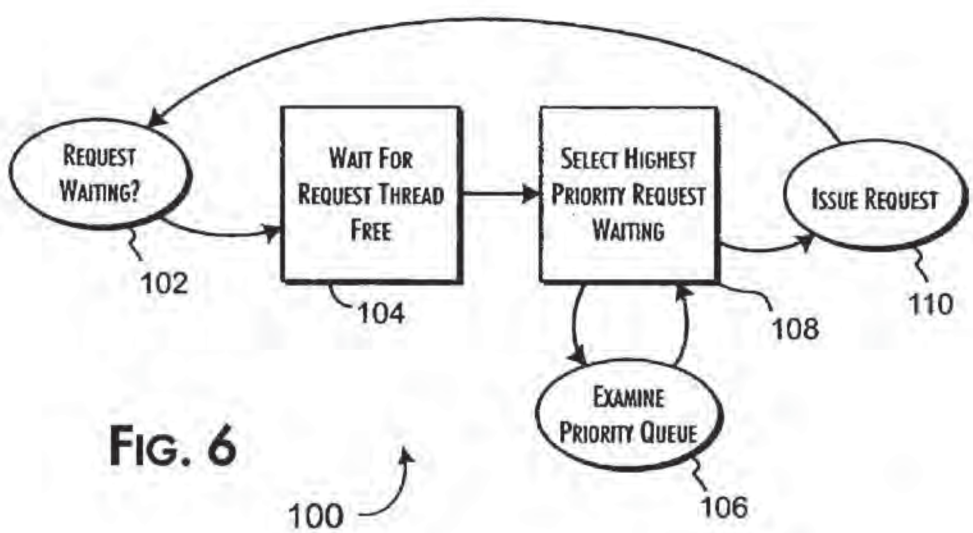
68
NETWORK DATA PACKET
MICRO PROCESSOR
GRAPHICS MEMORY
64
PARCEL DATA
62
70
L1 DATA CACHE
66
60

FIG. 4



82
Navigation Events
Determine Viewable Parcels For Update
Examine QuadTrees
80
84
Request Parcel Downloads
Issue Priority Requests
Network
88
Trim QuadTree Nodes
94
Receive Parcel
Store Parcel To QuadTree Node
90
92

FIG. 5



Request Waiting?
Wait For Request Thread Free
Select Highest Priority Request Waiting
Issue Request
102
104
108
110
Examine Priority Queue
106
100

FIG. 6

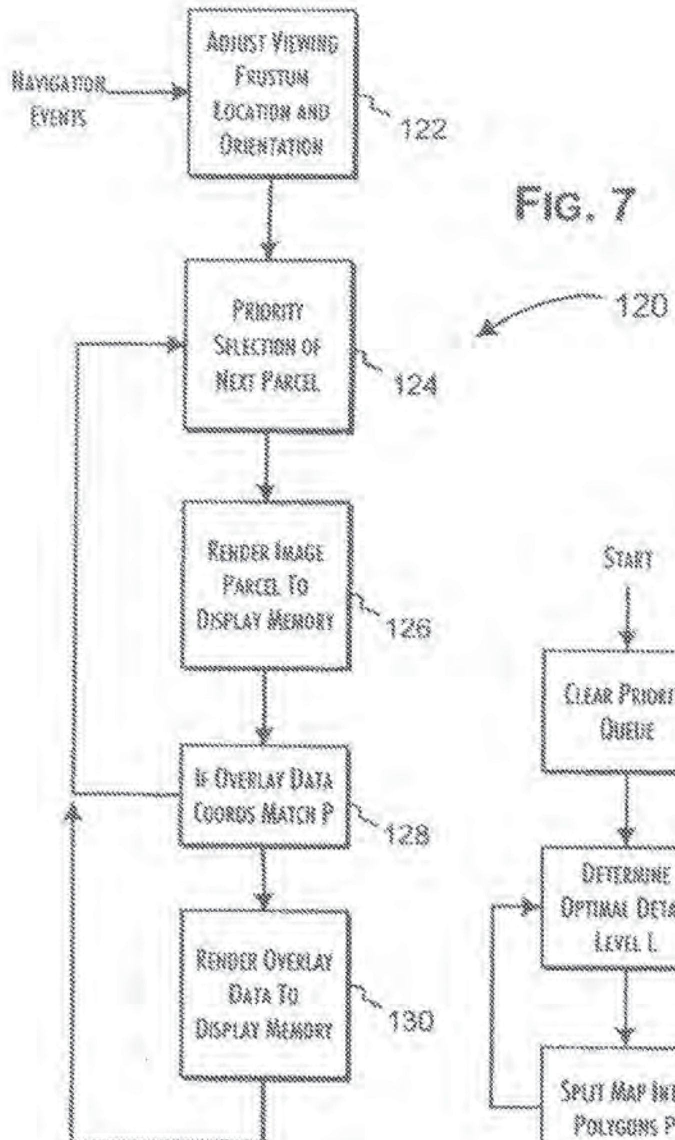

NAVIGATOR EVENTS
ADJUST VIEWING FRUSTUM LOCATION AND ORIENTATION
122
120
PRIORITY SELECTION OF NEXT PARCEL
124
RENDER IMAGE PARCEL TO DISPLAY MEMORY
126
IF OVERLAY DATA COORDS MATCH P
128
RENDER OVERLAY DATA TO DISPLAY MEMORY
130

FIG. 7

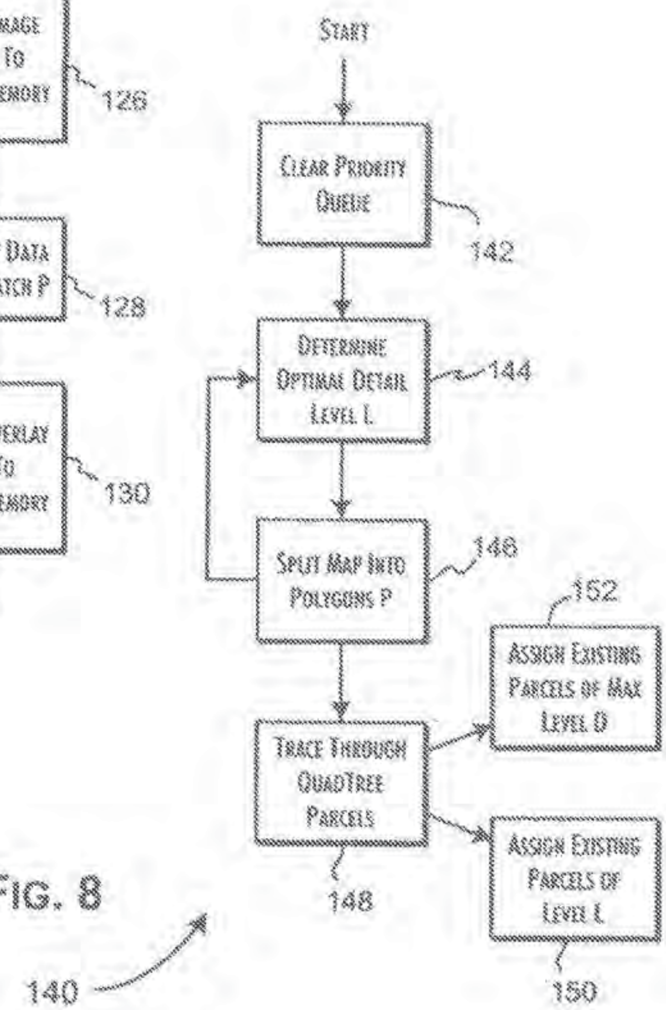

START
CLEAR PRIORITY QUEUE
142
DETERMINE OPTIMAL DETAIL LEVEL L
144
SPLIT MAP INTO POLYGONS P
146
TRACE THROUGH QUADTREE PARCELS
148
ASSIGN EXISTING PARCELS OF MAX LEVEL D
152
ASSIGN EXISTING PARCELS OF LEVEL L
150
140

FIG. 8

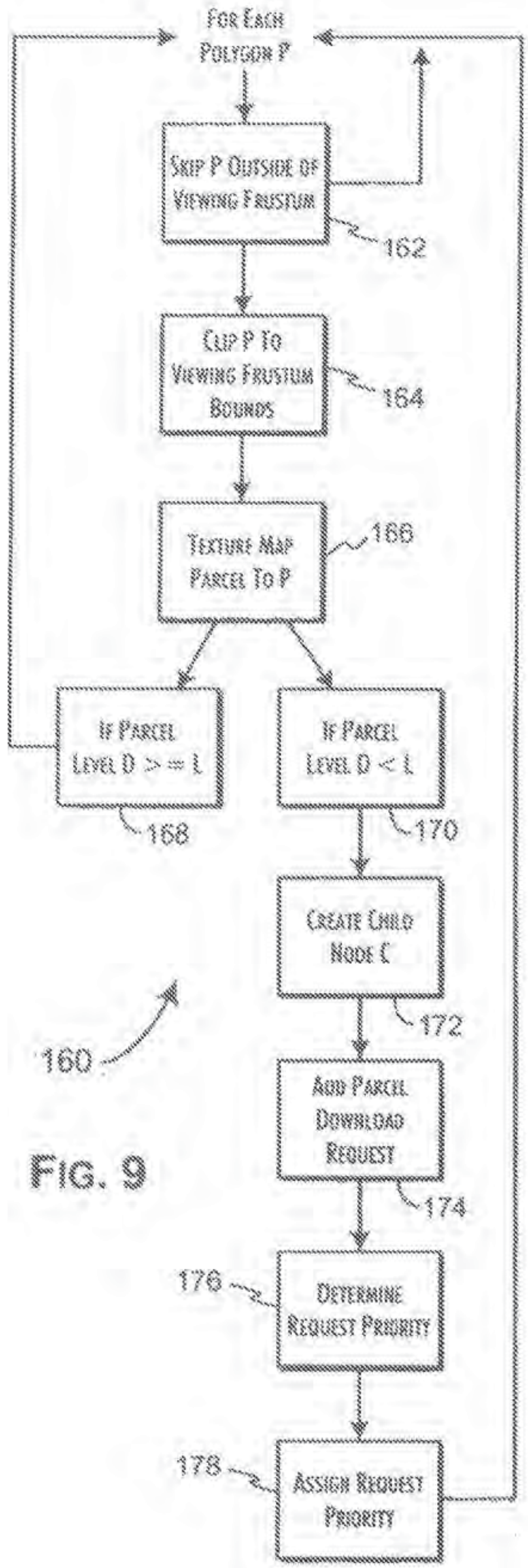

For Each Polygon P
Skip P Outside of Viewing Frustum
162
Clip P To Viewing Frustum Bounds
164
Texture Map Parcel To P
166
If Parcel Level D >= L
168
If Parcel Level D < L
170
Create Child Node C
172
Add Parcel Download Request
174
176
Determine Request Priority
178
Assign Request Priority
160

FIG. 9

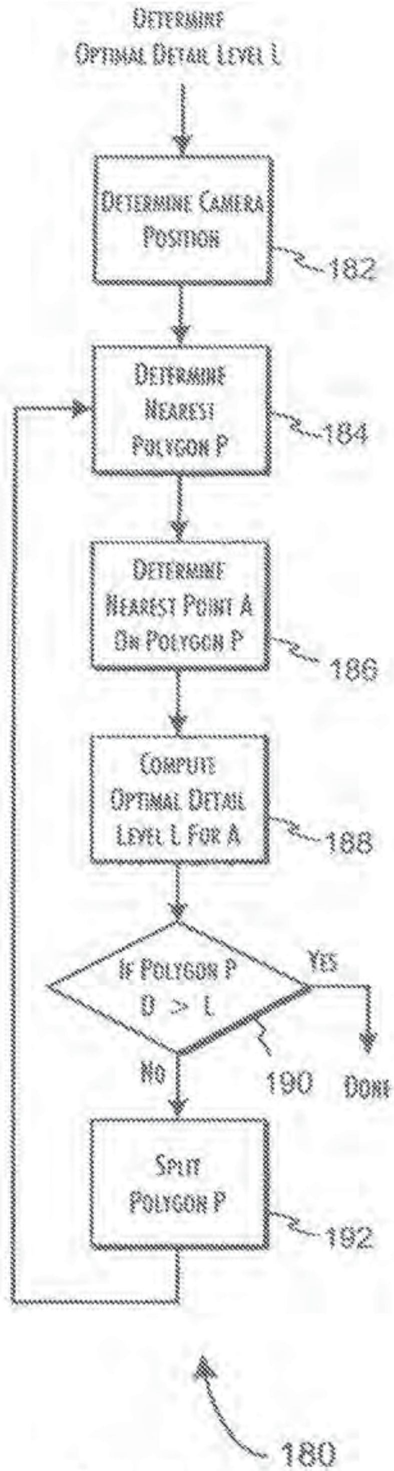

Determine Optimal Detail Level L
Determine Camera Position
182
Determine Nearest Polygon P
184
Determine Nearest Point A On Polygon P
186
Compute Optimal Detail Level L For A
188
If Polygon P D > L
Yes
No
190
Done
Split Polygon P
192
180

FIG. 10

OPTIMIZED IMAGE DELIVERY OVER LIMITED BANDWIDTH COMMUNICATION CHANNELS

PRIORITY CLAIMS/RELATED APPLICATIONS

[0001] This application is a continuation in part of and claims priority under 35 USC 120 to U.S. patent application Ser. No. 12/619,643 filed on Nov. 16, 2009 which in turn in a continuation of and claims priority under 35 USC 120 to U.S. patent application Ser. No. 10/035,987 filed on Dec. 24, 2001 and entitled "Optimized image delivery over limited bandwidth communication channels" (that now issued on Jan. 5, 2010 as U.S. Pat. No. 7,644,131) which in turn claims the benefit under 35 USC 119(*e*) of U.S. Provisional Application Nos. 60/258,488, 60/258,489, 60/258,465, 60/258,468, 60/258,466, and 60/258,467, all filed Dec. 27, 2000, all of which are incorporated herein by reference. The present application is also related to the co-pending application Ser. No. 10/035,981 entitled "System and Methods for Network Image Delivery with Dynamic Viewing Frustum Optimized for Limited Bandwidth Communication Channels, Levanon et al., filed on Dec. 24, 2001 (now U.S. Pat. No. 7,139,794 issued on Nov. 21, 2006 and which is assigned to the Assignee of the present Application.

FIELD

[0002] The disclosure is related to network based, image distribution systems and, in particular, to a system and methods for efficiently selecting and distributing image parcels through a narrowband or otherwise limited bandwidth communications channel to support presentation of high-resolution images subject to dynamic viewing frustums.

BACKGROUND

[0003] The Internet and or other network systems may provide a unique opportunity to transmit for example complex images, typically large scale bit-maps, particularly those approaching photo-realistic levels, over large area and or distances. In common application, the images may be geographic, topographic, and or other highly detailed maps. The data storage requirements and often proprietary nature of such images could be such that conventional interests may be to transfer the images on an as-needed basis.

[0004] In conventional fixed-site applications, the image data may be transferred over a relatively high-bandwidth network to client computer systems that in turn, may render the image. Client systems may typically implement a local image navigation system to provide zoom and or pan functions based on user interaction. As well recognized problem with such conventional systems could be that full resolution image presentation may be subject to the inherent transfer latency of the network. Different conventional systems have been proposed to reduce the latency affect by transmitting the image in highly compressed formats that support progressive resolution build-up of the image within the current client field of view. Using a transform compressed image transfer function increases the field of the image that can be transferred over a fixed bandwidth network in unit time. Progressive image resolution transmission, typically using a differential resolution method, permits an approximate image to be quickly presented with image details being continuously added over time.

[0005] Tzou, in U.S. Pat. No. 4,698,689, describes a two-dimensional data transform system that supports transmission of differential coefficients to represent an image. Subsequent transmitted coefficient sets are progressively accumulated with prior transmitted sets to provide a succeedingly refined image. The inverse-transform function performed by the client computer is, however, highly compute intensive. In order to simplify the transform implementation and further reduce the latency of presenting any portion of an approximate image, images are sub-divided into a regular array. This enables the inverse-transform function on the client, which is time-critical, to deal with substantially smaller coefficient data sets. The array size in Tzou is fixed, which leads to progressively larger coefficient data sets as the detail level of the image increases. Consequently, there is an inherently increasing latency in resolving finer levels of detail.

[0006] An image visualization system proposed by Yap et al., U.S. Pat. No. 6,182,114, overcomes some of the foregoing problems. The Yap et al. system also employs a progressive encoding transform to compress the image transfer stream. The transform also operates on a subdivided image, but the division is indexed to the encoding level of the transform. The encoded transform coefficient data sets are, therefore, of constant size, which supports a modest improvement in the algorithmic performance of the inverse transform operation required on the client.

[0007] Yap et al. adds utilization of client image panning or other image pointing input information to support a foveation-based operator to influence the retrieval order of the subdivided image blocks. This two-dimensional navigation information is used to identify a foveal region that is presumed to be the gaze point of a client system user. The foveation operator defines the corresponding image block as the center point of an ordered retrieval of coefficient sets representing a variable resolution image. The gaze point image block represents the area of highest image resolution, with resolution reduction as a function of distance from the gaze point determined by the foveation operator. This technique thus progressively builds image resolution at the gaze point and succeedingly outward based on a relatively compute intensive function. Shifts in the gaze point can be responded to with relative speed by preferentially retrieving coefficient sets at and near the new foveal region.

[0008] Significant problems remain in permitting the convenient and effective use of complex images by many different types of client systems, even with the improvements provided by the various conventional systems. In particular, the implementation of conventional image visualization systems is generally unworkable for smaller, often dedicated or embedded, clients where use of image visualization would clearly be beneficial. Conventional approaches effectively presume that client systems have an excess of computing performance, memory and storage. Small clients, however, typically have restricted performance processors with possibly no dedicated floating-point support, little general purpose memory, and extremely limited persistent storage capabilities, particularly relative to common image sizes. A mobile computing device such as mobile phone, smart phone, tablet and or personal digital assistant (PDA) is a characteristic small client. Embedded, low-cost kiosk, automobile navigation systems and or Internet enabled/connected TV are other typical examples. Such systems are not readily capable, if at all, of performing complex, compute-intensive Fourier or wavelet transforms, particularly within a highly restricted memory address space.

[0009] As a consequence of the presumption that the client is a substantial computing system, conventional image visualization systems also presume that the client is supported by a complete operating system. Indeed, many expect and require an extensive set of graphics abstraction layers to be provided by the client system to support the presentation of the delivered image data. In general, these abstraction layers are conventionally considered required to handle the mapping of the image data resolution to the display resolution capabilities of the client system. That is, resolution resolved image data provided to the client is unconstrained by any limitation in the client system to actually display the corresponding image. Consequently, substantial processor performance and memory can be conventionally devoted to handling image data that is not or cannot be displayed.

[0010] Another problem is that small clients are generally constrained to generally to very limited network bandwidths, particularly when operating under wireless conditions. Such limited bandwidth conditions may exist due to either the direct technological constraints dictated by the use of a low bandwidth data channel or indirect constraints imposed on relatively high-bandwidth channels by high concurrent user loads. Cellular connected PDAs and webphones are examples of small clients that are frequently constrained by limited bandwidth conditions. The conventionally realizable maximum network transmission bandwidth for such small devices may range from below one kilobit per second to several tens of kilobits per second. While Yap et al. states that the described system can work over low bandwidth lines, little more than utilizing wavelet-based data compression is advanced as permitting effective operation at low communications bandwidths. While reducing the amount of data that must be carried from the server to the client is significant, Yap et al. simply relies on the data packet transfer protocols to provide for an efficient transfer of the compressed image data. Reliable transport protocols, however, merely mask packet losses and the resultant, sometimes extended, recovery latencies. When such covered errors occur, however, the aggregate bandwidth of the connection is reduced and the client system can stall waiting for further image data to process.

[0011] Consequently, there remains a need for an image visualization system that can support small client systems, place few requirements on the supporting client hardware and software resources, and efficiently utilize low to very low bandwidth network connections.

SUMMARY

[0012] Thus, a general purpose of the present invention is to provide an efficient system and methods of optimally presenting image data on client systems with potentially limited processing performance, resources, and communications bandwidth.

[0013] This is achieved in the present invention by providing for the retrieval of large-scale images over network communications channels for display on a client device by selecting an update image parcel relative to an operator controlled image viewpoint to display via the client device. A request is prepared for the update image parcel and associated with a request queue for subsequent issuance over a communications channel. The update image parcel is received from the communications channel and displayed as a discrete portion of the predetermined image. The update image parcel optimally has a fixed pixel array size, is received in a single and or plurality of network data packets, and were the fixed pixel array may be constrained to a resolution less than or equal to the resolution of the client device display.

[0014] An advantage of the present invention is that both image parcel data requests and the rendering of image data are optimized to address the display based on the display resolution of the client system.

[0015] Another advantage of the present invention is that the prioritization of image parcel requests is based on an adaptable parameter that minimizes the computational complexity of determining request prioritization and, in turn, the progressive improvement in display resolution within the field of view presented on a client display.

[0016] A further advantage of the present invention is that the client software system requires relatively minimal client processing power and storage capacity. Compute intensive numerical calculations are minimally required and image parcel data is compactly stored in efficient data structures. The client software system is very small and easily downloaded to conventional computer systems or embedded in conventional dedicated function devices, including portable devices, such as PDAs, tablets and webphones.

[0017] Still another advantage of the present invention is that image parcel data requests and presentation can be readily optimized to use low to very low bandwidth network connections. The software system of the present invention provides for re-prioritization of image parcel data requests and presentation in circumstances where the rate of point-of-view navigation exceeds the data request rate.

[0018] Yet another advantage of the present invention is that image parcel data rendering is performed without requiring any complex underlying hardware or software display subsystem. The client software system of the present invention includes a bit-map rendering engine that draws directly to the video memory of the display, thus placing minimal requirements on any underlying embedded or disk operating system and display drivers. Complex graphics and animation abstraction layers are not required.

[0019] Still another advantage of the present invention is that image parcel block compression is used to obtain fixed size transmission data blocks. Image parcel data is recoverable from transmission data using a relatively simple client decompression algorithm. Using fixed size transmission data blocks enables image data parcels to be delivered to the client in bounded time frames.

[0020] A yet further advantage of the present invention is that multiple data forms can be transferred to the client software system for concurrent display. Array overlay data, correlated positionally to the image parcel data and generally insensitive to image parcel resolution, can be initially or progressively provided to the client for parsing and parallel presentation on a client display image view.

BRIEF DESCRIPTION OF THE DRAWINGS

[0021] These and other advantages and features of the present invention will become better understood upon consideration of the following detailed description of the invention when considered in connection with the accompanying drawings, in which like reference numerals designate like parts throughout the figures thereof, and wherein:

[0022] FIG. **1** depicts a preferred system environment within which various embodiments of the present invention can be utilized;

[0023] FIG. **2** is a block diagram illustrating the preparation of image parcel and overlay data set that are to be stored by and served from a network server system in accordance with a preferred embodiment of the present invention;

[0024] FIG. **3** is a block diagram of a client system image presentation system constructed in accordance with a preferred embodiment of the present invention;

[0025] FIG. **4** provides a data block diagram illustrating an optimized client image block processing path constructed in accordance with a preferred embodiment of the present invention;

[0026] FIG. **5** is a process flow diagram showing a main processing thread implemented in a preferred embodiment of the present invention;

[0027] FIG. **6** provides a process flow diagram showing a network request thread implemented in a preferred embodiment of the present invention;

[0028] FIG. **7** provides a process flow diagram showing a display image rendering thread implemented in a preferred embodiment of the present invention;

[0029] FIG. **8** provides a process flow diagram showing the parcel map processing performed preliminary to the rendering of image data parcels in accordance with a preferred embodiment of the present invention;

[0030] FIG. **9** provides a process flow diagram detailing the rendering and progressive prioritization of image parcel data download requests in accordance with a preferred embodiment of the present invention; and

[0031] FIG. **10** provides a process flow diagram detailing the determination of an optimal detail level for image parcel presentation for a current viewing frustum in accordance with a preferred embodiment of the present invention.

DETAILED DESCRIPTION OF ONE OR MORE EMBODIMENTS

[0032] The preferred operational environment **10** of the present invention is generally shown in FIG. **1**. A network server system **12**, operating as a data store and server of image data, is responsive to requests received through a communications network, such as the Internet **14** generally and various tiers of internet service providers (ISPs) including a wireless connectivity provider **16**. Client systems, including conventional workstations and personal computers **18** and smaller, typically dedicated function devices often linked through wireless network connections, such as PDAs, webphones **20**, and automobile navigation systems, source image requests to the network server **12**, provide a client display and enable image navigational input by a user of the client system. Alternately, a dedicated function client system **20** may be connected through a separate or plug-in local network server **22**, preferably implementing a small, embedded Web server, to a fixed or removable storage local image repository **24**. Characteristically, the client system **18**, **20** displays are operated at some fixed resolution generally dependent on the underlying display hardware of the client systems **18**, **20**.

[0033] The image navigation capability supported by the present invention encompasses a viewing frustum placed within a three-dimensional space over the imaged displayed on the client **18**, **20**. Client user navigational inputs are supported to control the x, y lateral, rotational and z height positioning of the viewing frustum over the image as well as the camera angle of incidence relative to the plane of the image. To effect these controls, the software implemented on the client systems **18**, **20** supports a three-dimensional transform of the image data provided from the server **12**, **22**.

[0034] In accordance with the preferred embodiments of the present invention, as generally illustrated in FIG. **2**, a network image server system **30** stores a combination of source image data **32** and source overlay data **34**. The source image data **32** is typically high-resolution bit-map raster map and or satellite imagery of geographic regions, which can be obtained from commercial suppliers. The overlay image data **34** is typically a discrete data file providing image annotation information at defined coordinates relative to the source image data **32**. In the preferred embodiments of the present invention, image annotations include, for example, street, building and landmark names, as well as representative 2 and 3D objects, graphical icons, decals, line segments, and or text and or other characters, graphics and or other media.

[0035] The network image server system **30** preferably pre-processes the source image data **32** and or source overlay data **34** to forms preferred for storage and serving by the network server **12**, **22**. The source image data **32** is preferably pre-processed to obtain a series K.sub.1-N of derivative images of progressively lower image resolution. The source image data **32**, corresponding to the series image K.sub.0, is also subdivided into a regular array such that each resulting image parcel of the array has for example a 64 by 64 pixel resolution where the image data has a color or bit per pixel depth of 16 bits, which represents a data parcel size of 8K bytes. The resolution of the series K.sub.1-N of derivative images is preferably related to that of the source image data **32** or predecessor image in the series by a factor of four. The array subdivision is likewise related by a factor of four such that each image parcel is of a fixed 8K byte size.

[0036] In the preferred embodiment of the present invention, the image parcels are further compressed and stored by the network server **12**, **22**. The preferred compression algorithm may implements for example a fixed 4:1 compression ratio such that each compressed and stored image parcel has a fixed 2K byte size. The image parcels are preferably stored in a file of defined configuration such that any image parcel can be located by specification of a K.sub.D, X, Y value, representing the image set resolution index D and corresponding image array coordinate.

[0037] In other implementations, the image array dimensions (which as 64×64 above) may be powers of two so that the image array can be used in texture mapping efficiently. To accommodate different data parcel size than the 2 KByte associated with 64×64 pixel parcel dimension described above and other communication protocol and overhead requirements, to accommodate transmission through other than a 3 KByte per second transmission channel, the present invention may use larger compression ratios that takes, for example, a 128×128 or 256×256 pixel parcel dimension and compresses it to meet the 3 KByte per second transmission channel, or other communication bandwidth used to stream the parcel.

[0038] The system may also accommodate different and larger data parcel sizes as transmission protocols, compression ratio achieved and micro-architectures of the client computers change. For purposes above, the data content was a pixel array representing image data. Where the data parcel content is vector, text or other data that may subject to different client system design factors, other parcel sizes may be used. Furthermore, the parcel sizes can be different between the server and the client. For example the server may create parcels or hold parcels, for streaming with 256×256 pixel parcel dimension and the client my render them as 64×64. In addition, parcels sizes on different servers may vary from one server to another and from the client side rendering. In the system, each grid is treated as a sparse data array that can be progressively revised to increase the resolution of the grid and thereby the level of detail presented by the grid.

[0039] The source overlay data **34** is preferably pre-processed **36** into either an open XML format, such as the Geography Markup Language (GML), which is an XML based encoding standard for geographic information developed by the OpenGIS Consortium (OGC; www.opengis.org), or a proprietary binary representation. The XML/GML representation is preferred as permitting easier interchange between different commercial entities, while the binary representation is preferred as more compact and readily transferable to a client system **18**, **20**. In both cases, the source overlay data **34** is pre-processed to contain the annotation data preferably in a resolution independent form associated with a display coordinate specification relative to the source image data **32**. The XML, GML or binary overlay data may be compressed prior to storage on the network server **12**, **22**.

[0040] The preferred architecture **40** of a client system **18**, **20**, for purposes of implementing the present invention, is shown in FIG. **3**. The architecture **40** is preferably implemented by a software plug-in or application executed by the client system **18**, **20** and that utilizes basic software and hardware services provided by the client system **18**, **20**. A parcel request client **42** preferably implements an HTML client that supports HTML-based interactions with the server **12**, **22** using the underlying network protocol stack and hardware network interface provided by the client systems **18**, **20**. A central parcel processing control block **44** preferably implements the client process and control algorithms. The control block **44** directs the transfer of received image parcels and XML/GML/binary overlay data to a local parcel data store **46**. Local parcel data store **46** may also act for example as local cache weather the entire data or part of it is in dynamic and/or static cache. Preferably image data parcels are stored in conventional quad-tree data structures, where tree nodes of depth D correspond to the stored image parcels of a derivative image of resolution KD. The XML/GML/binary overlay data is preferably stored as a data object that can be subsequently read by an XML/GML/binary parser implemented as part of the control block **44**.

[0041] The control block **44** is also responsible for decompressing and directing the rendering of image parcels to a local display by a rendering engine **48**. Preferably, the rendering engine **48** writes to the video memory of the underlying client display hardware relying on only generic graphics acceleration hardware capabilities and may take advantage of more advanced graphics acceleration hardware when available in the client system **18**, **20**. In general, the relied on capabilities include bit-bit and related bit-oriented functions that are readily supported by current conventional display controller hardware. The rendering engine **48** is optimized to perform image parcel texture mapping without reliance on complex floating point operations, permitting even relatively simple processors to efficiently execute the rendering engine **48**. The rendering engine **48** may take advantage of floating point operations when available in the client system **18**, **20**.

[0042] Changes in the viewing frustum are determined from user input navigation commands by a frustum navigation block **50**. In the preferred embodiments of the present invention, the input navigation controls are modeled for three-dimensional fly-over navigation of the displayed image. The navigation controls support point-of-view rotation, translation, attitude, and altitude over the displayed image. The effective change in viewing frustum as determined by the frustum navigation block **50** is provided to the control block **44**.

[0043] The control block **44**, based in part on changes in the viewing frustum, determines the ordered priority of image parcels to be requested from the server **12**, **22** to support the progressive rendering of the displayed image. The image parcel requests are placed in a request queue **52** for issuance by the parcel request client **42**. Preferably, the pending requests are issued in priority order, thereby dynamically reflecting changes in the viewing frustum with minimum latency.

[0044] In various implementations of the parcel processing, each data parcel is independently processable by the client system **18**,**20**, which is enabled by the selection and server-side processing used to prepare a parcel for transmission, thus providing for on-demand real-time parcel processing and creation on the server side for streaming based on the client request and not only for pre-processed parcel creation for retrieval for streaming from the server. Thus, the system can use both pre-processed parcels on the server and on-demand real-time creation of such parcels on the server side for streaming to the client.

[0045] An optimal image parcel data flow **60**, as configured for use in the preferred embodiments of the present invention, is shown in FIG. **4**. Preferably, the TCP/IP network protocol is used to deliver image parcels to the clients **18**, **20**. For the preferred embodiments, where network bandwidth is limited or very limited, entire image parcels are preferably delivered in corresponding data packets. This preference maximizes data delivery while avoiding the substantial latency and processing overhead of managing image parcel data split over multiple network packets. Thus, a 2K byte compressed image parcel **62** is delivered as the data payload of a TCP/IP packet **64**. Uncompressed, the 8K byte image parcel **62** is recognized as part of the present invention as being within the nominally smallest L1 data cache **66** size of conventional microprocessors **68**. By ensuring that an uncompressed image parcel fits within the L1 cache, the texture map rendering algorithm can execute with minimum memory management overhead, thus optimally utilizing the processing capability of the microprocessor **68**. Additionally, the writing of video data as a product of the rendering algorithm is uniform, thereby improving the apparent video stability of the display to the user.

[0046] The client architecture **40** preferably executes in multiple process threads, with additional threads being utilized for individual network data request transactions. As shown in FIG. **5**, an image parcel management process **80** implements a loop that determines image parcels subject to update **82** and creates corresponding image parcel download requests **84**. Navigation events that alter the viewing frustum are considered in part to determine the current field of view. The quad-tree data structures are examined **86** to identify viewable image parcels of higher resolution than currently available in the parcel data store **46**.

[0047] A pool of image request threads is preferably utilized to manage the image parcel download operations. In the preferred embodiments of the present invention, a pool of four network request threads is utilized. The number of pool threads is determined as a balance between the available system resources and the network response latency, given the available bandwidth of the network connection. Empirically, for many wireless devices, four concurrent threads are able to support a relatively continuous delivery of image data parcels to the client **20** for display processing. As image parcels are progressively identified for download, a free request thread is employed to issue 88 a corresponding network request to the server **12**, **22**. When a network response is received, the corresponding thread recovers **90** the image parcel data. The received image parcel is then stored **92** in a corresponding quad-tree data structure node.

[0048] For small clients **20**, the available memory for the parcel data store **46** is generally quite restricted. In order to make optimal use of the available memory, only currently viewable image parcels are subject to download. Where the size of the parcel data store **46** is not so restricted, this constraint can be relaxed. In either case, a memory management process **94** runs to monitor use of the parcel data store **46** and selectively remove image parcels to free memory for newly requested image parcels. Preferably, the memory management process **94** operates to preferentially remove image parcels that are the furthest from the current viewing frustum and that have the highest data structure depth. Preferably child node image parcels are always removed before a parent node parcel is removed.

[0049] A preferred network request management process **100** is shown in FIG. **6**. The process **100** waits **102** on the existence of a download request in the priority request queue **52**. The process **100** then waits on a network request pool thread to become free **104**. When a network request thread becomes available, the process **100** examines **106** all of the pending requests in the priority request queue **52** and selects **108** the request with the highest assigned priority. Thus, sequentially enqueued requests can be selectively issued out of order based on an independently assigned request priority. The request is then issued **110** and the request management process **100** leaves the request thread waiting on a network response.

[0050] FIG. **7** presents a preferred display management process **120**. Event driven user navigation information is evaluated **122** to determine a current viewing frustum location and orientation within a three-dimensional space relative to the displayed image. An algorithmic priority selection **124** of a next image parcel to render is then performed. The selected image parcel is then rendered **126** to the display memory **70**. The rendering operation preferably performs a texture map transform of the parcel data corresponding to the current viewing frustum location and orientation. The overlay data is then parsed or is pre-parsed to determine **128** whether the image coordinates of any overlay annotation correspond to the current image parcel location. If the coordinates match, the overlay annotation is rendered **130** to the video display memory **70**. The process **120** then continues with the next selection **124** of an image parcel to render, subject to any change in the viewing frustum location and orientation.

[0051] A preferred implementation of the selection **124** and rendering **126** of image parcels in accordance with the present invention is detailed in FIGS. **8** through **10**. Referring first to FIG. **8**, any outstanding requests in the priority request queue **52** are preferably cleared **142** in response to a change in the viewing frustum location and orientation. The effective altitude of the viewing frustum and or the resolution of the client display are then used as a basis for determining an optimal level of detail L that will be displayed. The detail level L value operates as a floor defining the maximum resolution K.sub.L of image data that can be effectively viewed on the client display given the location and or orientation of the viewing frustum. Constraining image parcel requests to the resolution range K.sub.N to K.sub.L, where K.sub.N is the lowest resolution derivative image stored by the network server **12**, **22**, prevents the download and processing of image parcels that cannot provide any perceptible improvement in the displayed image.

[0052] As part of the recursive evaluation of the optimal level of detail L, the image display space is progressively split **146** by four to one reductions into polygons. The quad-tree data structures holding existing image parcel data in the parcel data store **46** are concurrently traced **148** to establish a correspondence with the polygon map. Where the trace of a quad-tree data structure completes **150** to a node index of L for a polygon P, the node corresponding image parcel is associated with polygon P. The polygon P will not be further subdivided and no higher resolution image parcels will be requested for any portion of the image within the area represented by polygon P. Where the trace reaches a maximum node index of D for a polygon P' **152**, where N.ltoreq.D<L and N is the index of the lowest resolution derivative image stored by the network server **12**, **22**, the image parcel associated with the node is associated with the polygon P'. This polygon P' will be subject to further subdivision and progressive requests for image parcels of higher resolution up to the detail level L.

[0053] Referring now to FIG. **9**, a display image is then rendered **160** beginning with the maximum depth polygons previously found. Iterating over the set of maximum depth polygons, any polygons outside of the viewing frustum are skipped **162**. Polygons that are at least partially visible are clipped to the applicable bounds of the viewing frustum **164**. The polygon corresponding image parcel data is then texture mapped **166** into the polygon corresponding coordinates of the video memory **70**. If the node index depth of the rendered image parcel is at least equal to the prior determined optimal detail level L **168**, the iteration over the polygons P continues.

[0054] Where the node index depth is less than the optimal detail level L **170**, the polygon P' is subdivided into four polygons and correspondingly represented by the creation of four child nodes within the associated quad-tree data structure **172**. Four image parcel download requests are then created **174**.

[0055] The download priority associated with each request is determined **176** by execution of a function S that operates on a 2D polygon argument P and returns a real number representing the request priority. The function argument P is a list of real (x, y) coordinates of the vertices of the current polygon in screen coordinates after being clipped to fit within the current viewing frustum. That is, the function S works over general polygons in a two-dimensional space, whose vertices are specified by the series $\{(x(1),y(1)),(x(2),y(2)), \ldots ,(x(n),y(n))\}$. The argument P vertices sent to S represent the position of the vertices composing each of the polygons, after being clipping to the viewing frustum, viewable within the display space having the fixed resolution [xRes, yRes]. Thus, the clipped polygons are all within the rectangle $[0, xRes]\times[0, yRes]$.

[0056] In execution of the function S, each of the P coordinates is first transformed by linear mapping of the screen coordinate space to the square [−1,1].times.[−1,1] by the operation x(i):=(x(i)−xRes/2)/(xRes/2); y(i)=(y(i)−yRes/2)/(yRes/2). The x and y coordinate values of each vertex (x(i), y(i)) for i=1 to n) are then transformed by the function T(a)=sgn(a)*pow(.vertline.a.vertline., d), where the control parameter d is a constant in the range (0,1], or equivalently the interval 0<dltoreq.1. The function S then returns a real value that is equal to the area covered by the argument polygon P vertices subject to the applied coordinate transformation. Thus, the accumulated priority for any image parcel pending download is the sum of the values of returned by the function S for each of the viewable polygons that require some part of the image parcel as the source data for texture map rendering of the polygon. The priority operation of the request queue **52** is such that download requests will be issued preferentially for image parcels with the largest priority value.

[0057] In accordance with the preferred embodiments of the present invention, the value of the control parameter d can be adjusted to ultimately affect the behavior of the function S in determining the download request priority. In general, image parcels with lower resolution levels will accumulate greater priority values due to the larger number of polygons that may use a given low resolution image parcel as a rendering data source. Such lower resolution image parcels are therefore more likely to be preferentially downloaded. In accordance with the present invention, this generally assures that a complete image of at least low resolution will be available for rendering.

[0058] The control parameter d, as applied in execution of the function S, well as the area distortion produced by the projection transform also influences the value returned by the function S such that relatively higher-resolution image parcels near the image view point will occasionally achieve a higher priority than relatively remote and partially viewed image parcels of lower resolution. Using values smaller than 1 for the control parameter d results in requests with a higher priority for parcels covering areas near the focal point of the viewer, which is presumed to be the center point of the display space, relative to requests for parcels further from the center point in absolute terms and of the same resolution depth D. Thus, in accordance with the present invention, the priority assigned to image parcel requests effectively influences the order of requests based on the relative contribution of the image parcel data to the total display quality of the image. Empirically, a value of 0.35 for the control parameter d for small screen devices, such as PDAs and webphones has been found to produce desirable results.

[0059] The computed priorities of each of the four newly created image parcel requests are then assigned **178** and the requests are enqueued in the priority request queue **52**. The next polygon P is then considered in the loop of the image parcel rendering process **160**.

[0060] The preferred algorithm **180** for determining the detail level L value for a given viewing frustum is shown in FIG. **10**. In accordance with the present invention, the optimal detail level L is effectively the limit at which the resolution of image parcel data functionally exceeds the resolution of the client display. Preferably, to determine the optimal detail level L, the viewpoint or camera position of the viewing frustum is determined **182** relative to the displayed image. A nearest polygon P of depth D is then determined **184** from the effective altitude and attitude of the viewpoint. The nearest point A of the polygon P is then determined **186**. The point A may be within the interior or an edge of the polygon P, though most likely be located at a vertex of the polygon P.

[0061] The optimum level of detail L at point A is then computed **188** as the base-4 logarithm of the number of pixels on the screen that would be covered by a single pixel from an image parcel of the lowest resolution K.sub.-N image, which is the quad-tree root image and corresponds to an image area covering the entire image map. The point A optimal detail level L is preferably computed analytically from the local value of the Jacobian of the projective transform used to transform the three dimensional image coordinate space to screen coordinates, evaluated at the point A.

[0062] Where the depth D of the polygon P is greater than the depth of the computed optimal level of detail L, the detail level L is taken as the optimal detail level L **190**. Thus, through the process **140**, an image parcel or corresponding section of the closest resolution image parcel associated with a parent node in the quad-tree data structure relative to the depth level L will be used as the texture for rendering the polygon P. Conversely, if the depth D is less than that of the optimal detail level L, the polygon P is effectively split into quadrants and the optimal level of detail is reevaluated. The process **180** thus continues iteratively until the optimal detail level L is found.

[0063] Thus, a system and methods of optimally presenting image data on client systems with potentially limited processing performance, resources, and communications bandwidth have been described. While the present invention has been described particularly with reference to the communications and display of geographic image data, the present invention is equally applicable to the efficient communications and display of other high resolution information.

[0064] In the process implemented by the system described above, data parcels may be selected for sequential transmission based on a prioritization of the importance of the data contained. The criteria for the importance of a particular data parcel may be defined as suitable for particular applications and may directly relate to the presentation of image quality, provision of a textual overlay of a low-quality image to quickly provide a navigational orientation, or the addition of topography information at a rate or timing different from the rate of image quality improvement. Thus, image data layers reflecting navigational cues, text overlays, and topography can be composed into data packets for transmission subject to prioritizations set by the server alone and not based on the client system and interactively influenced by the actions and commands provided by the user of the client system. However this also may be influenced based on the nature and type of the client system, and interactively influenced by the actions and commands provided by the user of the client system (FIG. **5**).

[0065] In view of the above description of the preferred embodiments of the present invention, many modifications and variations of the disclosed embodiments will be readily appreciated by those of skill in the art. It is therefore to be understood that, within the scope of the appended claims, the invention may be practiced otherwise than as specifically described above.

1. A method of retrieving large-scale images over network communications channels for display on a limited communication bandwidth computer device, said method comprising:

issuing, from a limited communication bandwidth computer device to a remote computer, a request for an update data parcel wherein the update data parcel is selected based on an operator controlled image viewpoint on the computer device relative to a predetermined image and the update data parcel contains data that is used to generate a display on the limited communication bandwidth computer device;

processing, on the remote computer, source image data to obtain a series K_{1-N} of derivative images of progressively lower image resolution and wherein series image K_0 being subdivided into a regular array wherein each resulting image parcel of the array has a predetermined pixel resolution wherein image data has a color or bit per pixel depth representing a data parcel size of a predetermined number of bytes, resolution of the series K_{1-N} of derivative images being related to that of the source image data or predecessor image in the series by a factor of two, and said array subdivision being related by a factor of two such that each image parcel being of a fixed byte size;

receiving said update data parcel from the data parcel stored in the remote computer over a communications channel; and

displaying on the limited communication bandwidth computer device using the update data parcel that is a part of said predetermined image, an image wherein said update data parcel uniquely forms a discrete portion of said predetermined image.

2. The method of claim **1**, wherein processing the source image data further comprises one of pre-processing the source image data on the remote computer and processing the source image data in real-time on-demand based on the request for the updated image parcel.

3. The method of claim **2**, wherein receiving the update data parcel over a communications channel further comprises streaming the update data parcel over a communications channel to the limited communication bandwidth computer device.

4. The method of claim **1**, wherein the limited communication bandwidth computer device further comprises one of a mobile computer system, a cellular computer system, an embedded computer system, a handheld computer system, a personal digital assistants and an internet-capable digital phone and a television.

5. The method of claim **1**, wherein a size of the data parcel on the remote computer is different from the update data parcel on the limited communication bandwidth computer device.

6. The method of claim **1**, wherein processing the source image data further comprises queuing the update data parcels on the remote computer based on an importance of the update data parcel as determined by the remote computer.

7. The method of claim **1**, wherein the processing further comprises compressing each data parcel and storing each data parcel on the remote computer in a file of defined configuration such that a data parcel can be located by specification of a K_D, X, Y value that represents the data set resolution index D and corresponding image array coordinate.

8. A display system for displaying a large-scale image retrieved over a limited bandwidth communications channel, said display system comprising:

a display of defined screen resolution for displaying a defined image;

a memory providing for the storage of a plurality of image parcels displayable over respective portions of a mesh corresponding to said defined image;

a communications channel interface supporting the retrieval of a defined data parcel over a limited bandwidth communications channel;

a processor coupled between said display, memory and communications channel interface, said processor operative to select said defined data parcel, retrieve said defined data parcel via said limited bandwidth communications channel interface for storage in said memory, and render said defined data parcel over a discrete portion of said mesh to provide for a progressive resolution enhancement of said defined image on said display; and

wherein a remote computer coupled to the limited bandwidth communications channel, delivers the defined data parcel wherein delivering the defined data parcel further comprises processing source image data to obtain a series K_{1-N} of derivative images of progressively lower image resolution and wherein series image K_0 being subdivided into a regular array wherein each resulting image parcel of the array has a predetermined pixel resolution wherein image data has a color or bit per pixel depth representing a data parcel size of a predetermined number of bytes, resolution of the series K_{1-N} of derivative images being related to that of the source image data or predecessor image in the series by a factor of two, and said array subdivision being related by a factor of two such that each image parcel being of a fixed byte size.

9. The display system of claim **8**, wherein processing the source image data further comprises one of pre-processing the source image data on the remote computer and processing the source image data in real-time on-demand based on the request for the updated image parcel.

10. The display system of claim **9**, wherein receiving the update data parcel over a communications channel further comprises streaming the update data parcel over a communications channel to the limited communication bandwidth computer device.

11. The display system of claim **8**, wherein the limited communication bandwidth computer device further comprises one of a mobile computer system, a cellular computer system, an embedded computer system, a handheld computer system, a personal digital assistants and an internet-capable digital phone and a television.

12. The display system of claim **8**, wherein a size of the data parcel on the remote computer is different from the update data parcel on the limited communication bandwidth computer device.

13. The display system of claim **8**, wherein processing the source image data further comprises queuing the update data parcels on the remote computer based on an importance of the update data parcel as determined by the remote computer.

14. The display system of claim **8**, wherein the processing may further comprises compressing each data parcel and storing each data parcel on the remote computer in a file of defined configuration such that a data parcel can be located by specification of a K_D, X, Y value that represents the data set resolution index D and corresponding image array coordinate.

15. A remote computer for delivering large-scale images over network communications channels for display on a limited communication bandwidth computer device that has a display system for displaying a large-scale image retrieved over a limited bandwidth communications channel, a display of defined screen resolution for displaying a defined image, a memory providing for the storage of a plurality of image parcels displayable over respective portions of a mesh corresponding to said defined image, a communications channel interface supporting the retrieval of a defined data parcel over a limited bandwidth communications channel and a processor coupled between said display, memory and communications channel interface, said processor operative to select said defined data parcel, retrieve said defined data parcel via said limited bandwidth communications channel interface for storage in said memory, and render said defined data parcel over a discrete portion of said mesh to provide for a progressive resolution enhancement of said defined image on said display, the remote computer comprises:

a parcel processing unit that processes a piece of source image data and delivers the defined data parcel to the limited communication bandwidth computer device; and

wherein the parcel processing unit further comprises a parcel processing control that processes source image data to obtain a series K_{1-N} of derivative images of progressively lower image resolution and wherein series image K_0 being subdivided into a regular array wherein each resulting image parcel of the array has a predetermined pixel resolution wherein image data has a color or bit per pixel depth representing a data parcel size of a predetermined number of bytes, resolution of the series K_{1-N} of derivative images being related to that of the source image data or predecessor image in the series by a factor of two, and said array subdivision being related by a factor of two such that each image parcel being of a fixed byte size.

16. The remote computer of claim **15**, wherein processing the source image data further comprises one of pre-processing the source image data on the remote computer and processing the source image data in real-time on-demand based on the request for the updated image parcel.

17. The remote computer of claim **16**, wherein receiving the update data parcel over a communications channel further comprises streaming the update data parcel over a communications channel to the limited communication bandwidth computer device.

18. The remote computer of claim **15**, wherein the limited communication bandwidth computer device further comprises one of a mobile computer system, a cellular computer system, an embedded computer system, a handheld computer system, a personal digital assistants and an internet-capable digital phone and a television.

19. The remote computer of claim **15**, wherein a size of the data parcel on the remote computer is different from the update data parcel on the limited communication bandwidth computer device.

20. The remote computer of claim **15**, wherein processing the source image data further comprises queuing the update data parcels on the remote computer based on an importance of the update data parcel as determined by the remote computer.

21. The remote computer of claim **15**, wherein processing further comprises compressing each data parcel and storing each data parcel on the remote computer in a file of defined configuration such that a data parcel can be located by specification of a K_D, X, Y value that represents the data set resolution index D and corresponding image array coordinate.

* * * * *

EXHIBIT G

To the First Amended Complaint

US009253239B2

(12) **United States Patent**
Levanon et al.

(10) **Patent No.: US 9,253,239 B2**
(45) **Date of Patent: *Feb. 2, 2016**

(54) **OPTIMIZED IMAGE DELIVERY OVER LIMITED BANDWIDTH COMMUNICATION CHANNELS**

(71) Applicant: **BRADIUM TECHNOLOGIES LLC**, Suffern, NY (US)

(72) Inventors: **Isaac Levanon**, Raanana (IL); **Yonatan Lavi**, Raanana (IL)

(73) Assignee: **BRADIUM TECHNOLOGIES LLC**, Suffern, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **14/547,148**

(22) Filed: **Nov. 19, 2014**

(65) **Prior Publication Data**

US 2015/0180928 A1 Jun. 25, 2015

Related U.S. Application Data

(63) Continuation of application No. 13/027,929, filed on Feb. 15, 2011, now Pat. No. 8,924,506, which is a continuation-in-part of application No. 12/619,643, filed on Nov. 16, 2009, now Pat. No. 7,908,343, which

(Continued)

(51) **Int. Cl.**

G06F 15/16	(2006.01)
H04L 29/06	(2006.01)

(Continued)

(52) **U.S. Cl.**
CPC ***H04L 65/602*** (2013.01); ***G06F 3/1454*** (2013.01); ***G06T 3/4092*** (2013.01); ***H04L 67/42*** (2013.01); *G09G 2340/02* (2013.01); *G09G 2350/00* (2013.01)

(58) **Field of Classification Search**
CPC G06T 3/4092; G09G 2340/02; G06F 3/14
USPC 709/202, 203, 217, 230, 246, 247; 345/625; 382/232, 305
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,682,869 A *	7/1987	Itoh et al.	....	358/426.12
4,972,319 A	11/1990	Delorme		

(Continued)

FOREIGN PATENT DOCUMENTS

EP	1070290 B1	1/2001
WO	99/41675	8/1999

OTHER PUBLICATIONS

Potmesil Maps Alive with dArnaud Declaration (all pages).

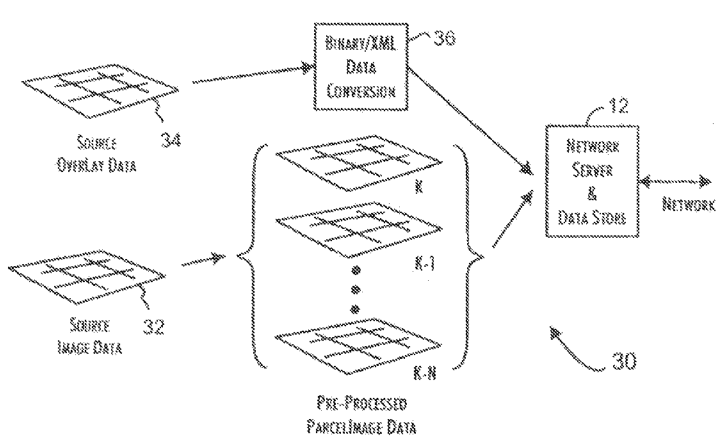
(Continued)

Primary Examiner — David Lazaro
(74) *Attorney, Agent, or Firm* — Anatoly S. Weiser, Esq.; Techlaw LLP

(57) **ABSTRACT**

Large-scale images are retrieved over network communications channels for display on a client device by selecting an update image parcel relative to an operator controlled image viewpoint to display via the client device. A request is prepared for the update image parcel and associated with a request queue for subsequent issuance over a communications channel. The update image parcel is received from the communications channel and displayed as a discrete portion of the predetermined image. The update image parcel optimally has a fixed pixel array size, is received in a single and or plurality of network data packets, and were the fixed pixel array may be constrained to a resolution less than or equal to the resolution of the client device display.

25 Claims, 5 Drawing Sheets



Related U.S. Application Data

is a continuation of application No. 10/035,987, filed on Dec. 24, 2001, now Pat. No. 7,644,131.

(60) Provisional application No. 60/258,465, filed on Dec. 27, 2000, provisional application No. 60/258,466, filed on Dec. 27, 2000, provisional application No. 60/258,467, filed on Dec. 27, 2000, provisional application No. 60/258,468, filed on Dec. 27, 2000, provisional application No. 60/258,488, filed on Dec. 27, 2000, provisional application No. 60/258,489, filed on Dec. 27, 2000.

(51) **Int. Cl.**
G06F 3/14 (2006.01)
G06T 3/40 (2006.01)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,179,638	A	1/1993	Dawson et al.	
5,263,136	A	11/1993	DeAguiar et al.	
5,321,520	A	6/1994	Inga et al.	
5,559,936	A *	9/1996	Poulter et al.	345/428
5,613,051	A *	3/1997	Iodice et al.	345/428
5,682,441	A	10/1997	Ligtenberg et al.	
5,760,783	A	6/1998	Migdal	
5,798,770	A	8/1998	Baldwin	
5,940,117	A	8/1999	Hassan et al.	
5,987,256	A	11/1999	Wu et al.	
5,995,903	A *	11/1999	Smith et al.	701/470
6,118,456	A	9/2000	Cooper	
6,167,442	A *	12/2000	Sutherland et al.	709/217
6,182,114	B1	1/2001	Yap et al.	
6,212,301	B1 *	4/2001	Warner et al.	382/232
6,246,797	B1 *	6/2001	Castor et al.	382/232
6,285,317	B1 *	9/2001	Ong	342/357.57
6,314,452	B1 *	11/2001	Dekel et al.	709/203
6,326,965	B1 *	12/2001	Castelli et al.	345/420
6,345,279	B1 *	2/2002	Li et al.	
6,346,938	B1 *	2/2002	Chan et al.	345/419
6,397,259	B1 *	5/2002	Lincke et al.	709/236
6,449,639	B1 *	9/2002	Blumberg	709/217
6,496,189	B1 *	12/2002	Yaron et al.	345/428
6,525,732	B1 *	2/2003	Gadh et al.	345/428
6,608,628	B1 *	8/2003	Ross et al.	345/619
6,608,933	B1 *	8/2003	Dowell et al.	382/232
6,625,309	B1 *	9/2003	Li et al.	382/173
6,650,998	B1	11/2003	Rutledge et al.	
6,704,024	B2 *	3/2004	Robotham et al.	345/581
6,704,791	B1 *	3/2004	Harris	709/231
6,711,297	B1 *	3/2004	Chang et al.	382/240
6,754,365	B1 *	6/2004	Wen et al.	382/100
6,801,665	B1 *	10/2004	Atsumi et al.	382/239
6,882,755	B2 *	4/2005	Silverstein et al.	382/282
6,898,311	B2 *	5/2005	Whitehead	382/166
6,970,604	B1 *	11/2005	Chai	382/240
7,908,343	B2 *	3/2011	Levanon et al.	709/217
8,924,506	B2 *	12/2014	Levanon et al.	709/217

OTHER PUBLICATIONS

Michalson Declaration Appendices P-EE (all pages).
Michalson Declaration with Appendices A-O (all pages).
Michael Potmesil, "Maps Alive: Viewing Geospatial Information on the WWW", Computer Networks and ISDN Systems vol. 29, issues 8-13, pp. 1327-1342.
Peter Lindstrom, David Koller, William Ribarsky, Larry F. Hodges, Augusto OP den Bosch, Nickk Faust, "An Integrated Global GIS and Visual Simulation System" by P. Lindstrom et al., Tech. Rep. GIT-GVU-97-07, Mar. 1997 (all pages).
Curriculum Vitae of William R. Michalson (all pages).
Hanan Samet, The Design and Analysis of Spatial Data Structures, University of Maryland (1989) (all pages).
B. Fuller and I. Richer, The Magic Project: From Vision to Reality, IEEE Network May/Jun. 1996, pp. 15-25.
International Telegraph and Telephone Consultative Committee ("CCITT") Recommendation T.81, Sep. 1992 (all pages).
Ken Cabeen & Peter Gent, Image Compression and the Discrete Cosine Transform (all pages).
M. Antonini, Image Coding Using Wavelet Transform , IEEE Transactions on Image Processing, vol. 1, No. 2, Apr. 1992 (all pages).
Lance Williams, Pyramidal Parametrics, Computer Graphics, vol. 17, No. 3, Jul. 1983, (all pages).
OpenGL Standard Version 1.1, Mar. 1997, available: https://www.opengl.org/documentation/specs/version1.1/glspec1.1/node84.html#SECTION00681100000000000000 , (all pages).
H. Hoppe, Progressive Meshes, SIGGRAPH '96: Proceedings of the 23rd annual conference on computer graphics and interactive techniques, pp. 99-108.
Boris Rabinovich and Craig Gotsman, Visualization of Large Terrains in Resource-Limited Computing Environments (1997) (all pages).
User Datagram Protocol (UDP) (Windows CE 5.0, Microsoft, Available: https://msdn.microsoft.com/enus/library/ms885773.aspx [Accessed Apr. 28, 2015] (all pages).
OpenGL Standard Version 1.2.1, Apr. 1999, available: https://www.opengl.org/documentation/specs/version1.2/opengl1.2.1.pdf , (all pages).
George H. Forman and John Zahorjan, "The challenges ofmobile computing," Computer vol. 27, No. 4, pp. 38, 47 (Apr. 1994).
K. Brown and S. Singh, A Network Architecture for Mobile Computing, INFOCOM '96, Fifteenth Annual Joint Conference of the IEEE Computer Societies, Networking the Next Generation, Proceedings IEEE vol. 3, pp. 1388-139.
B. Kreller et al "UMTS: a middleware architecture and mobile API approach," Personal Communications, IEEE, vol. 5, No. 2, pp. 32-38 (Apr. 1998).
J. Hansen et al, "Real-time synthetic vision cockpit display for general aviation," AeroSense '99, International Society for Optics and Photonics, 1999, (all pages).
Theresa-Marie Rhyne, A Commentary on GeoVRML: A Tool for 3D Representation of GeoReferenced Data on the Web, International Journal of Geographic Information Sciences, issue 4 of vol. 13, 1999, (all pages).
Declaration of Prof. William R. Michalson (all pages).
Provisional Applications to which the 506 Patent claims priority (all pages).
Numbering of Claim Elements of Challenged Claims of the 506 Patent, (all pages).
Claim chart illustrating teachings of Potmesil (Ex. 1002) and Hornbacker (Ex. 1003) pertinent to elements of Challenged Claims (all pages).
Claim chart illustrating teachings of Rutledge (Ex. 1006), Ligtenberg (Ex. 1005), and Cooper (Ex. 1007) pertinent to elements of Challenged Claims (all pages).
Claim chart illustrating teachings of Rutledge (Ex. 1006), Ligtenberg (Ex. 1005), Cooper (Ex. 1007), and Hassan (Ex. 1008) pertinent to elements of Challenged Claims (all pages).
Claim chart illustrating teachings of Fuller (App. E) and Hornbacker (Ex. 1003) pertinent to elements of Challenged Claims (all pages).
Claim chart illustrating teachings of Yap (App. J) and Rabinovich (App. R) pertinent to elements of Challenged Claims (all pages).

* cited by examiner

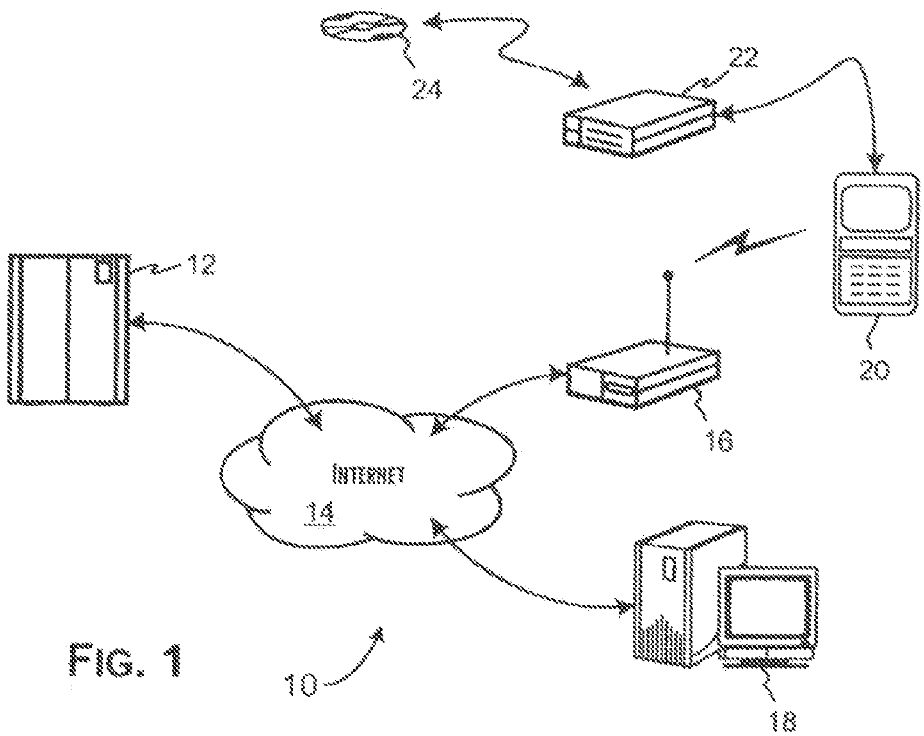

24
22
12
20
16
INTERNET
14
18
10

FIG. 1

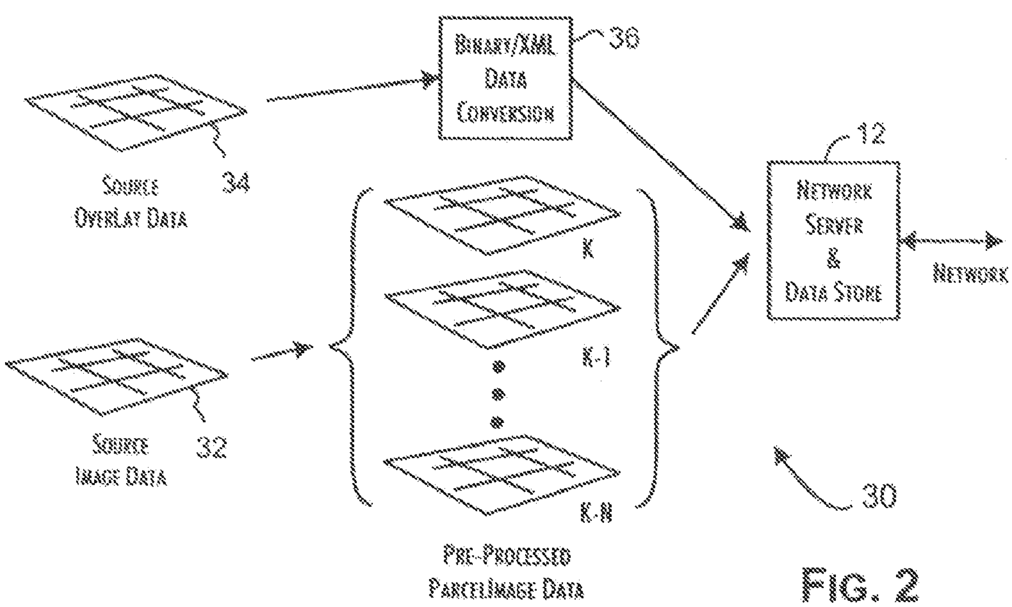

BINARY/XML
DATA
CONVERSION
36
12
SOURCE
OVERLAY DATA
34
NETWORK
SERVER
&
DATA STORE
NETWORK
K
K-1
K-N
SOURCE
IMAGE DATA
32
30
PRE-PROCESSED
PARCELIMAGE DATA

FIG. 2

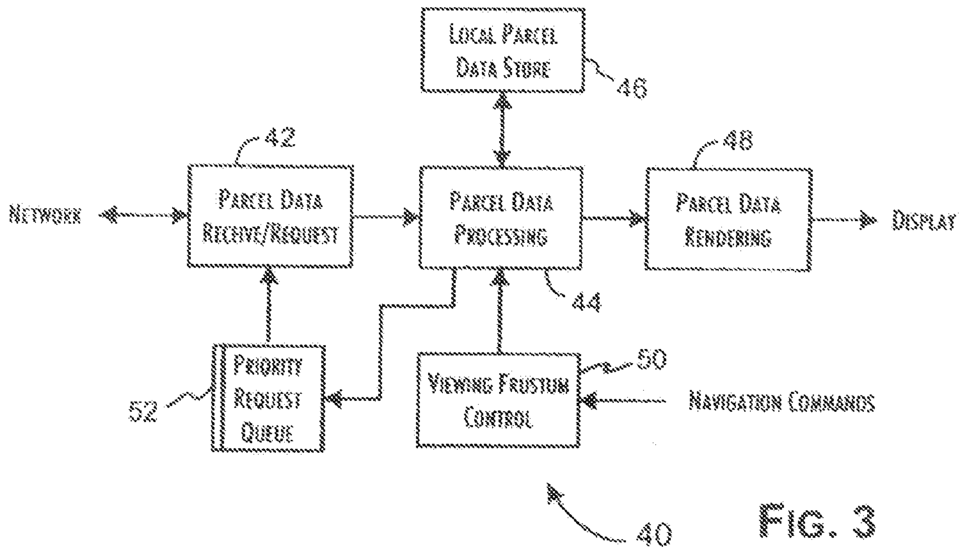

Local Parcel Data Store
46
42
48
Network
Parcel Data Receive/Request
Parcel Data Processing
Parcel Data Rendering
Display
44
50
Priority Request Queue
52
Viewing Frustum Control
Navigation Commands
40

FIG. 3

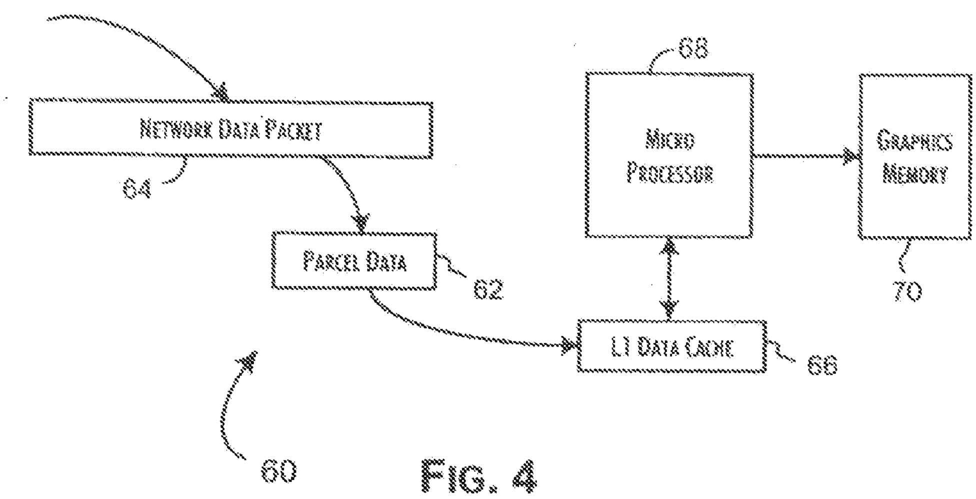
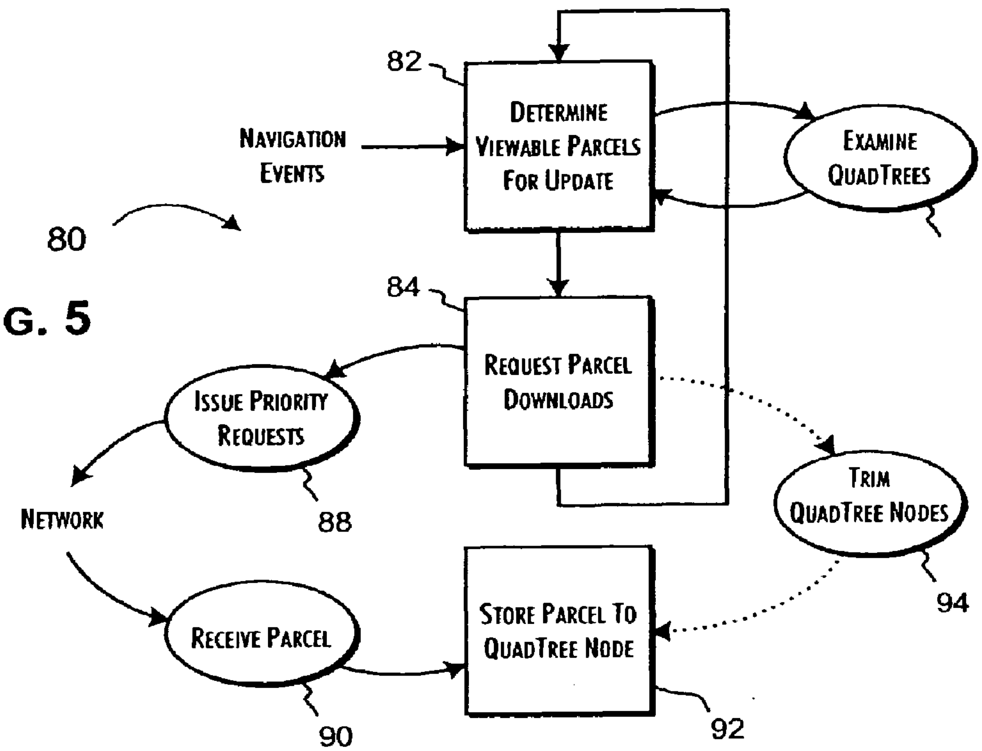
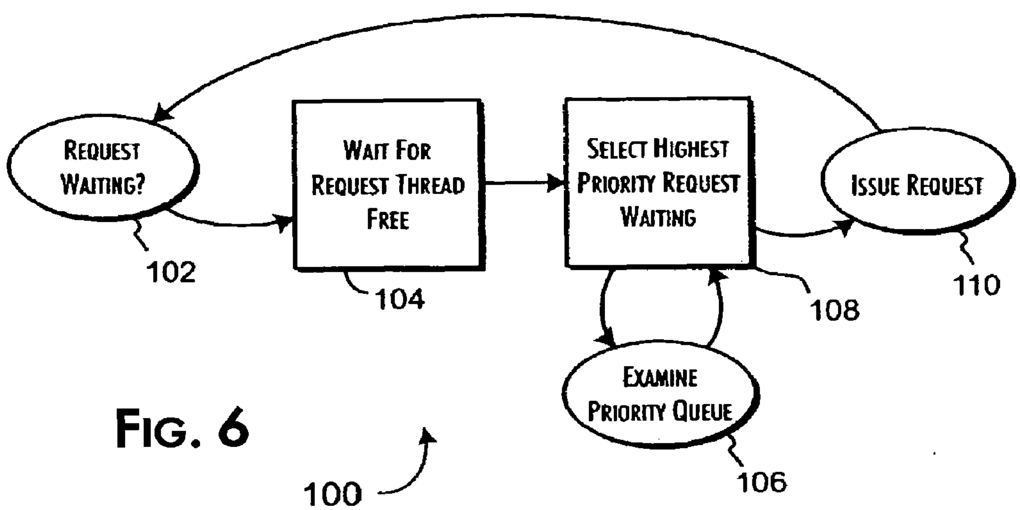

68
Network Data Packet
Micro Processor
Graphics Memory
64
Parcel Data
62
70
L1 Data Cache
66
60

FIG. 4



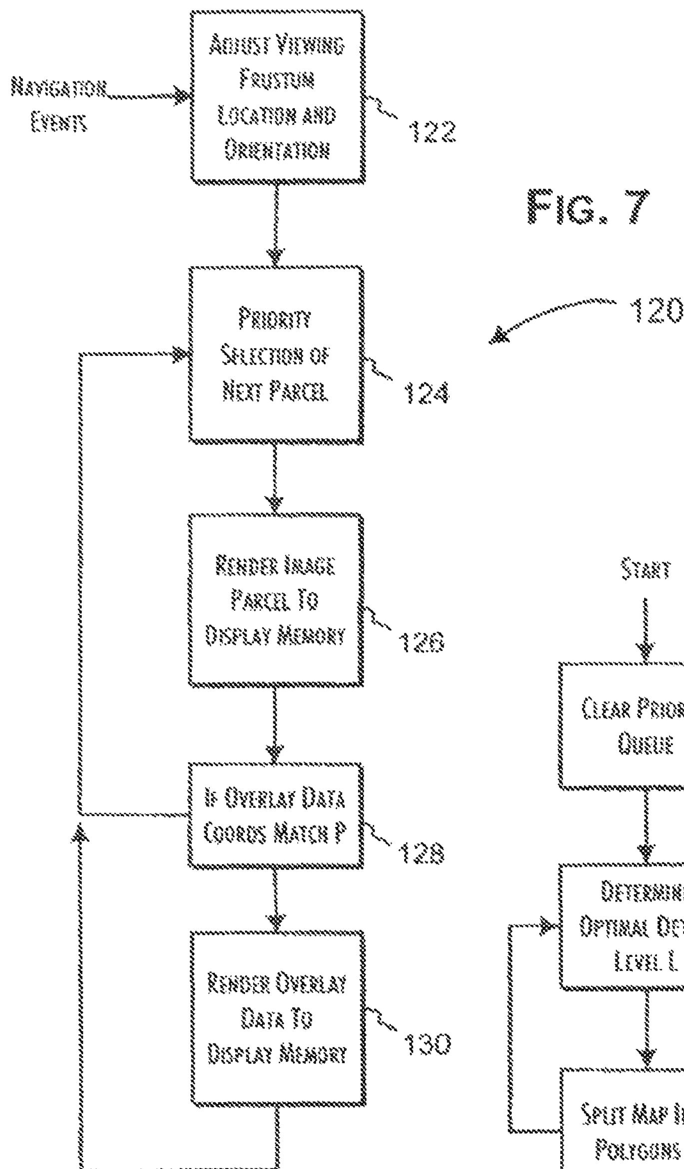
82
NAVIGATION EVENTS
DETERMINE VIEWABLE PARCELS FOR UPDATE
EXAMINE QUADTREES
80
84
REQUEST PARCEL DOWNLOADS
ISSUE PRIORITY REQUESTS
NETWORK
88
TRIM QUADTREE NODES
94
RECEIVE PARCEL
STORE PARCEL TO QUADTREE NODE
90
92

FIG. 5



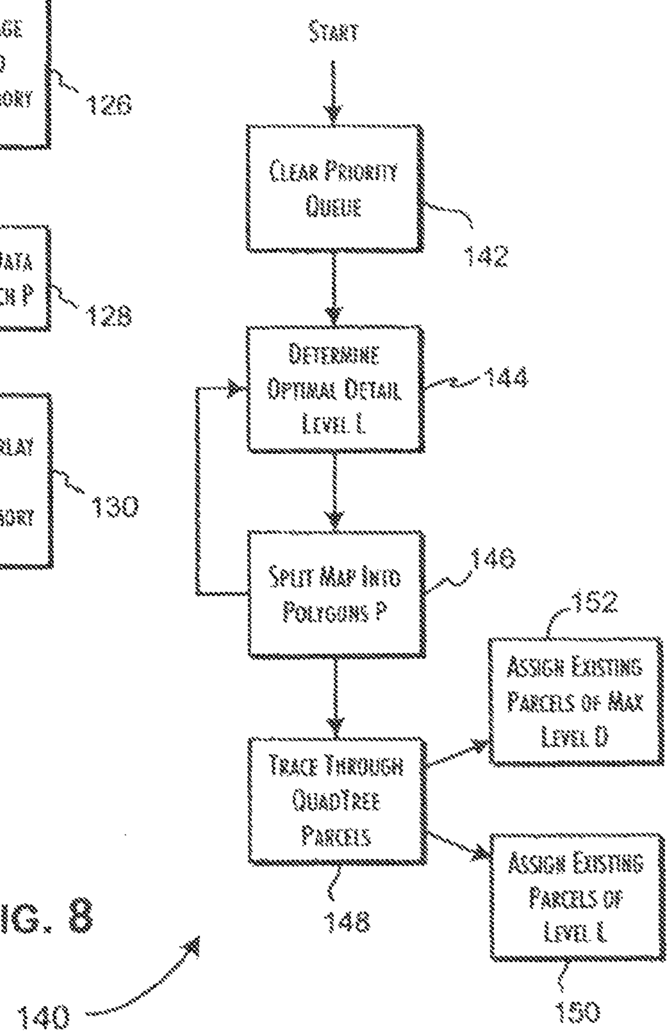
REQUEST WAITING?
WAIT FOR REQUEST THREAD FREE
SELECT HIGHEST PRIORITY REQUEST WAITING
ISSUE REQUEST
102
104
108
110
EXAMINE PRIORITY QUEUE
106
100

FIG. 6



Navigation Events
Adjust Viewing Frustum Location and Orientation
122
Priority Selection of Next Parcel
124
Render Image Parcel To Display Memory
126
If Overlay Data Coords Match P
128
Render Overlay Data To Display Memory
130
120

FIG. 7



Start
Clear Priority Queue
142
Determine Optimal Detail Level L
144
Split Map Into Polygons P
146
Trace Through QuadTree Parcels
148
Assign Existing Parcels of Max Level D
152
Assign Existing Parcels of Level L
150
140

FIG. 8

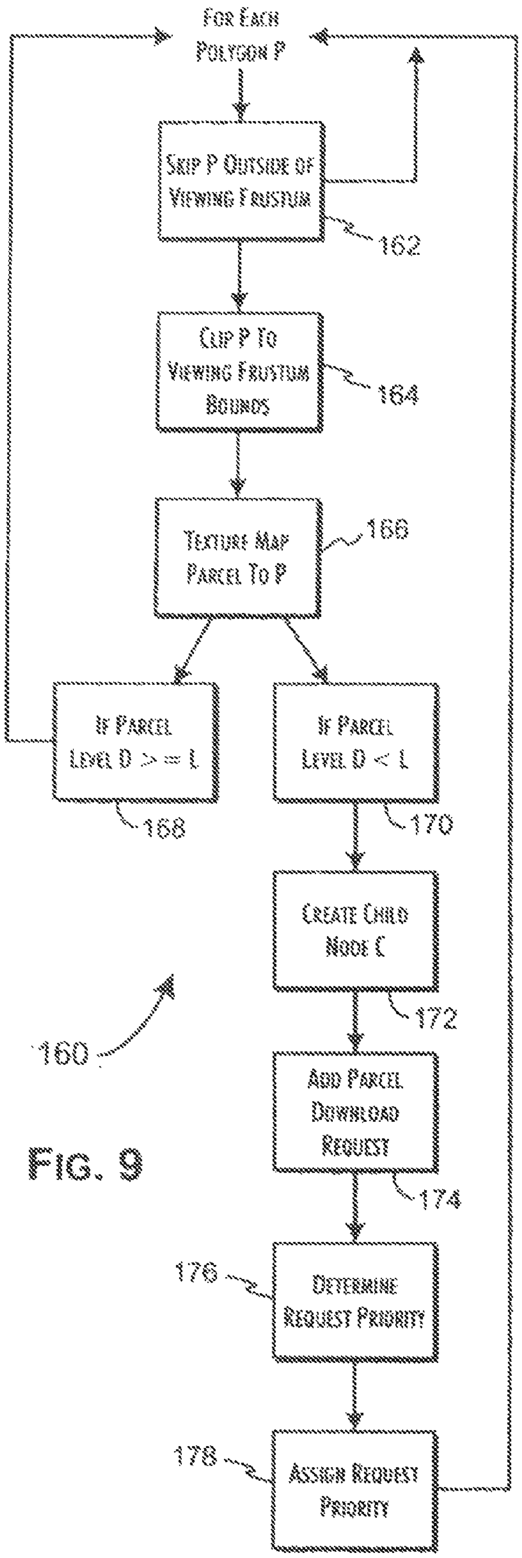

For Each Polygon P
Skip P Outside of Viewing Frustum
162
Clip P To Viewing Frustum Bounds
164
Texture Map Parcel To P
166
If Parcel Level D >= L
168
If Parcel Level D < L
170
Create Child Node C
172
Add Parcel Download Request
174
176
Determine Request Priority
178
Assign Request Priority
160

FIG. 9

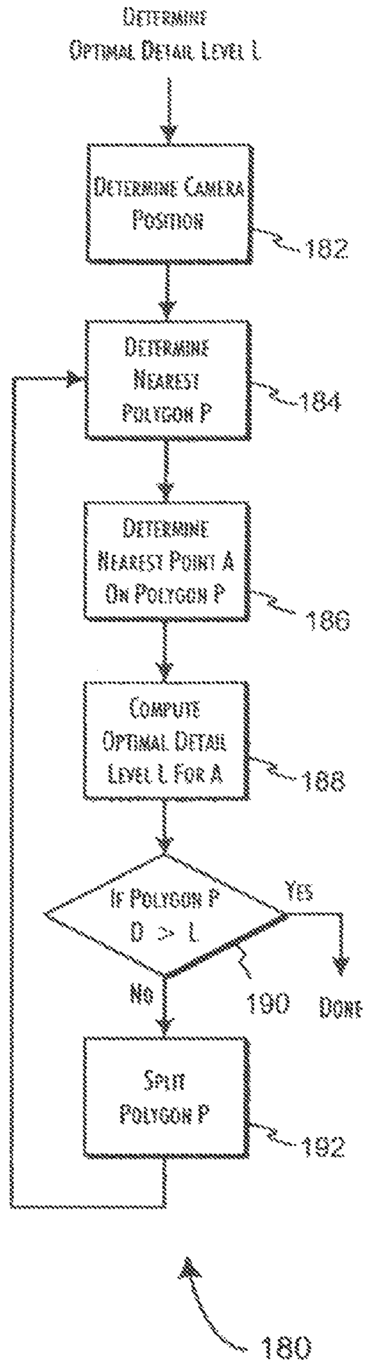

Determine Optimal Detail Level L
Determine Camera Position
182
Determine Nearest Polygon P
184
Determine Nearest Point A On Polygon P
186
Compute Optimal Detail Level L For A
188
If Polygon P D > L
Yes
No
190
Done
Split Polygon P
192
180

FIG. 10

OPTIMIZED IMAGE DELIVERY OVER LIMITED BANDWIDTH COMMUNICATION CHANNELS

PRIORITY CLAIMS/RELATED APPLICATIONS

This application is a continuation of and claims priority to U.S. patent application Ser. No. 13/027,929, entitled OPTIMIZED IMAGE DELIVERY OVER LIMITED BANDWIDTH COMMUNICATION CHANNELS, filed on 15 Feb. 2011, now allowed; which is a continuation in part of and claims priority under 35 USC 120 to U.S. patent application Ser. No. 12/619,643 filed on Nov. 16, 2009, now U.S. Pat. No. 7,908,343 which in turn is a continuation of and claims priority under 35 USC 120 to U.S. patent application Ser. No. 10/035,987 filed on Dec. 24, 2001 and entitled "Optimized image delivery over limited bandwidth communication channels" (that now issued on Jan. 5, 2010 as U.S. Pat. No. 7,644,131) which in turn claims the benefit under 35 USC 119(e) of U.S. Provisional Application Nos. 60/258,488, 60/258,489, 60/258,465, 60/258,468, 60/258,466, and 60/258,467, all filed Dec. 27, 2000, all of which are incorporated herein by reference.

FIELD

The disclosure is related to network based, image distribution systems and, in particular, to a system and methods for efficiently selecting and distributing image parcels through a narrowband or otherwise limited bandwidth communications channel to support presentation of high-resolution images subject to dynamic viewing frustums.

BACKGROUND

The Internet and or other network systems may provide a unique opportunity to transmit for example complex images, typically large scale bit-maps, particularly those approaching photo-realistic levels, over large area and or distances. In common application, the images may be geographic, topographic, and or other highly detailed maps. The data storage requirements and often proprietary nature of such images could be such that conventional interests may be to transfer the images on an as-needed basis.

In conventional fixed-site applications, the image data may be transferred over a relatively high-bandwidth network to client computer systems that in turn, may render the image. Client systems may typically implement a local image navigation system to provide zoom and or pan functions based on user interaction. As well recognized problem with such conventional systems could be that full resolution image presentation may be subject to the inherent transfer latency of the network. Different conventional systems have been proposed to reduce the latency affect by transmitting the image in highly compressed formats that support progressive resolution build-up of the image within the current client field of view. Using a transform compressed image transfer function increases the field of the image that can be transferred over a fixed bandwidth network in unit time. Progressive image resolution transmission, typically using a differential resolution method, permits an approximate image to be quickly presented with image details being continuously added over time.

Tzou, in U.S. Pat. No. 4,698,689, describes a two-dimensional data transform system that supports transmission of differential coefficients to represent an image. Subsequent transmitted coefficient sets are progressively accumulated with prior transmitted sets to provide a succeedingly refined image. The inverse-transform function performed by the client computer is, however, highly compute intensive. In order to simplify the transform implementation and further reduce the latency of presenting any portion of an approximate image, images are subdivided into a regular array. This enables the inverse-transform function on the client, which is time-critical, to deal with substantially smaller coefficient data sets. The array size in Tzou is fixed, which leads to progressively larger coefficient data sets as the detail level of the image increases. Consequently, there is an inherently increasing latency in resolving finer levels of detail.

An image visualization system proposed by Yap et al., U.S. Pat. No. 6,182,114, overcomes some of the foregoing problems. The Yap et al. system also employs a progressive encoding transform to compress the image transfer stream. The transform also operates on a subdivided image, but the division is indexed to the encoding level of the transform. The encoded transform coefficient data sets are, therefore, of constant size, which supports a modest improvement in the algorithmic performance of the inverse transform operation required on the client.

Yap et al. adds utilization of client image panning or other image pointing input information to support a foveation-based operator to influence the retrieval order of the subdivided image blocks. This two-dimensional navigation information is used to identify a foveal region that is presumed to be the gaze point of a client system user. The foveation operator defines the corresponding image block as the center point of an ordered retrieval of coefficient sets representing a variable resolution image. The gaze point image block represents the area of highest image resolution, with resolution reduction as a function of distance from the gaze point determined by the foveation operator. This technique thus progressively builds image resolution at the gaze point and succeedingly outward based on a relatively compute intensive function. Shifts in the gaze point can be responded to with relative speed by preferentially retrieving coefficient sets at and near the new foveal region.

Significant problems remain in permitting the convenient and effective use of complex images by many different types of client systems, even with the improvements provided by the various conventional systems. In particular, the implementation of conventional image visualization systems is generally unworkable for smaller, often dedicated or embedded, clients where use of image visualization would clearly be beneficial. Conventional approaches effectively presume that client systems have an excess of computing performance, memory and storage. Small clients, however, typically have restricted performance processors with possibly no dedicated floating-point support, little general purpose memory, and extremely limited persistent storage capabilities, particularly relative to common image sizes. A mobile computing device such as mobile phone, smart phone, tablet and or personal digital assistant (PDA) is a characteristic small client. Embedded, low-cost kiosk, automobile navigation systems and or Internet enabled I connected TV are other typical examples. Such systems are not readily capable, if at all, of performing complex, compute-intensive Fourier or wavelet transforms, particularly within a highly restricted memory address space.

As a consequence of the presumption that the client is a substantial computing system, conventional image visualization systems also presume that the client is supported by a complete operating system. Indeed, many expect and require an extensive set of graphics abstraction layers to be provided by the client system to support the presentation of the delivered image data. In general, these abstraction layers are conventionally considered required to handle the mapping of the image data resolution to the display resolution capabilities of the client system. That is, resolution resolved image data provided to the client is unconstrained by any limitation in the client system to actually display the corresponding image. Consequently, substantial processor performance and memory can be conventionally devoted to handling image data that is not or cannot be displayed.

Another problem is that small clients are generally constrained to generally to very limited network bandwidths, particularly when operating under wireless conditions. Such limited bandwidth conditions may exist due to either the direct technological constraints dictated by the use of a low bandwidth data channel or indirect constraints imposed on relatively high-bandwidth channels by high concurrent user loads. Cellular connected PDAs and webphones are examples of small clients that are frequently constrained by limited bandwidth conditions. The conventionally realizable maximum network transmission bandwidth for such small devices may range from below one kilobit per second to several tens of kilobits per second. While Yap et al. states that the described system can work over low bandwidth lines, little more than utilizing wavelet-based data compression is advanced as permitting effective operation at low communications bandwidths. While reducing the amount of data that must be carried from the server to the client is significant, Yap et al. simply relies on the data packet transfer protocols to provide for an efficient transfer of the compressed image data. Reliable transport protocols, however, merely mask packet losses and the resultant, sometimes extended recovery latencies. When such covered errors occur, however, the aggregate bandwidth of the connection is reduced and the client system can stall waiting for further image data to process.

Consequently, there remains a need for an image visualization system that can support small client systems, place few requirements on the supporting client hardware and software resources, and efficiently utilize low to very low bandwidth network connections.

SUMMARY

Thus, a general purpose of the present invention is to provide an efficient system and methods of optimally presenting image data on client systems with potentially limited processing performance, resources, and communications bandwidth.

This is achieved in the present invention by providing for the retrieval of large-scale images over network communications channels for display on a client device by selecting an update image parcel relative to an operator controlled image viewpoint to display via the client device. A request is prepared for the update image parcel and associated with a request queue for subsequent issuance over a communications channel. The update image parcel is received from the communications channel and displayed as a discrete portion of the predetermined image. The update image parcel optimally has a fixed pixel array size, is received in a single and or plurality of network data packets, and were the fixed pixel array may be constrained to a resolution less than or equal to the resolution of the client device display.

An advantage of the present invention is that both image parcel data requests and the rendering of image data are optimized to address the display based on the display resolution of the client system.

Another advantage of the present invention is that the prioritization of image parcel requests is based on an adaptable parameter that minimizes the computational complexity of determining request prioritization and, in turn, the progressive improvement in display resolution within the field of view presented on a client display.

A further advantage of the present invention is that the client software system requires relatively minimal client processing power and storage capacity. Compute intensive numerical calculations are minimally required and image parcel data is compactly stored in efficient data structures. The client software system is very small and easily downloaded to conventional computer systems or embedded in conventional dedicated function devices, including portable devices, such as PDAs, tablets and webphones.

Still another advantage of the present invention is that image parcel data requests and presentation can be readily optimized to use low to very low bandwidth network connections. The software system of the present invention provides for re-prioritization of image parcel data requests and presentation in circumstances where the rate of point-of-view navigation exceeds the data request rate.

Yet another advantage of the present invention is that image parcel data rendering is performed without requiring any complex underlying hardware or software display subsystem. The client software system of the present invention includes a bit-map rendering engine that draws directly to the video memory of the display, thus placing minimal requirements on any underlying embedded or disk operating system and display drivers. Complex graphics and animation abstraction layers are not required.

Still another advantage of the present invention is that image parcel block compression is used to obtain fixed size transmission data blocks. Image parcel data is recoverable from transmission data using a relatively simple client decompression algorithm. Using fixed size transmission data blocks enables image data parcels to be delivered to the client in bounded time frames.

A yet further advantage of the present invention is that multiple data forms can be transferred to the client software system for concurrent display. Array overlay data, correlated positionally to the image parcel data and generally insensitive to image parcel resolution, can be initially or progressively provided to the client for parsing and parallel presentation on a client display image view.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other advantages and features of the present invention will become better understood upon consideration of the following detailed description of the invention when considered in connection with the accompanying drawings, in which like reference numerals designate like parts throughout the figures thereof, and wherein:

FIG. **1** depicts a preferred system environment within which various embodiments of the present invention can be utilized;

FIG. **2** is a block diagram illustrating the preparation of image parcel and overlay data set that are to be stored by and served from a network server system in accordance with a preferred embodiment of the present invention;

FIG. **3** is a block diagram of a client system image presentation system constructed in accordance with a preferred embodiment of the present invention;

FIG. **4** provides a data block diagram illustrating an optimized client image block processing path constructed in accordance with a preferred embodiment of the present invention;

FIG. **5** is a process flow diagram showing a main processing thread implemented in a preferred embodiment of the present invention;

FIG. **6** provides a process flow diagram showing a network request thread implemented in a preferred embodiment of the present invention;

FIG. **7** provides a process flow diagram showing a display image rendering thread implemented in a preferred embodiment of the present invention;

FIG. **8** provides a process flow diagram showing the parcel map processing performed preliminary to the rendering of image data parcels in accordance with a preferred embodiment of the present invention;

FIG. **9** provides a process flow diagram detailing the rendering and progressive prioritization of image parcel data download requests in accordance with a preferred embodiment of the present invention; and

FIG. **10** provides a process flow diagram detailing the determination of an optimal detail level for image parcel presentation for a current viewing frustum in accordance with a preferred embodiment of the present invention.

DETAILED DESCRIPTION OF ONE OR MORE EMBODIMENTS

The preferred operational environment **10** of the present invention is generally shown in FIG. **1**. A network server system **12**, operating as a data store and server of image data, is responsive to requests received through a communications network, such as the Internet **14** generally and various tiers of internet service providers (ISPs) including a wireless connectivity provider **16**. Client systems, including conventional workstations and personal computers **18** and smaller, typically dedicated function devices often linked through wireless network connections, such as PDAs, webphones **20**, and automobile navigation systems, source image requests to the network server **12**, provide a client display and enable image navigational input by a user of the client system. Alternately, a dedicated function client system **20** may be connected through a separate or plug-in local network server **22**, preferably implementing a small, embedded Web server, to a fixed or removable storage local image repository **24**. Characteristically, the client system **18**, **20** displays are operated at some fixed resolution generally dependent on the underlying display hardware of the client systems **18**, **20**.

The image navigation capability supported by the present invention encompasses a viewing frustum placed within a three-dimensional space over the imaged displayed on the client **18**, **20**. Client user navigational inputs are supported to control the x, y lateral, rotational and z height positioning of the viewing frustum over the image as well as the camera angle of incidence relative to the plane of the image. To effect these controls, the software implemented on the client systems **18**, **20** supports a three-dimensional transform of the image data provided from the server **12**, **22**.

In accordance with the preferred embodiments of the present invention, as generally illustrated in FIG. **2**, a network image server system **30** stores a combination of source image data **32** and source overlay data **34**. The source image data **32** is typically high-resolution bit-map raster map and or satellite imagery of geographic regions, which can be obtained from commercial suppliers. The overlay image data **34** is typically a discrete data file providing image annotation information at defined coordinates relative to the source image data **32**. In the preferred embodiments of the present invention, image annotations include, for example, street, building and landmark names, as well as representative 2 and 3D objects, graphical icons, decals, line segments, and or text and or other characters, graphics and or other media.

The network image server system **30** preferably pre-processes the source image data **32** and or source overlay data **34** to forms preferred for storage and serving by the network server **12**, **22**. The source image data **32** is preferably pre-processed to obtain a series K.sub.1-N of derivative images of progressively lower image resolution. The source image data **32**, corresponding to the series image K.sub.O, is also subdivided into a regular array such that each resulting image parcel of the array has for example a 64 by 64 pixel resolution where the image data has a color or bit per pixel depth of 16 bits, which represents a data parcel size of 8K bytes. The resolution of the series K.sub.1-N of derivative images is preferably related to that of the source image data **32** or predecessor image in the series by a factor of four. The array subdivision is likewise related by a factor of four such that each image parcel is of a fixed 8K byte size.

In the preferred embodiment of the present invention, the image parcels are further compressed and stored by the network server **12**, **22**. The preferred compression algorithm may implements for example a fixed 4:1 compression ratio such that each compressed and stored image parcel has a fixed 2K byte size. The image parcels are preferably stored in a file of defined configuration such that any image parcel can be located by specification of a K.sub.D, X, Y value, representing the image set resolution index D and corresponding image array coordinate.

In other implementations, the image array dimensions (which as 64×64 above) may be powers of two so that the image array can be used in texture mapping efficiently. To accommodate different data parcel size than the 2 KByte associated with 64×64 pixel parcel dimension described above and other communication protocol and overhead requirements, to accommodate transmission through other than a 3 KByte per second transmission channel, the present invention may use larger compression ratios that takes, for example, a 128×128 or 256×256 pixel parcel dimension and compresses it to meet the 3 KByte per second transmission channel, or other communication bandwidth used to stream the parcel.

The system may also accommodate different and larger data parcel sizes as transmission protocols, compression ratio achieved and micro-architectures of the client computers change. For purposes above, the data content was a pixel array representing image data. Where the data parcel content is vector, text or other data that may subject to different client system design factors, other parcel sizes may be used. Furthermore, the parcel sizes can be different between the server and the client. For example the server may create parcels or hold parcels, for streaming with 256×256 pixel parcel dimension and the client my render them as 64×64. In addition, parcels sizes on different servers may vary from one server to another and from the client side rendering. In the system, each grid is treated as a sparse data array that can be progressively revised to increase the resolution of the grid and thereby the level of detail presented by the grid.

The source overlay data **34** is preferably pre-processed **36** into either an open XML format, such as the Geography Markup Language (GML), which is an XML based encoding standard for geographic information developed by the OpenGIS Consortium (OGC; www.opengis.org), or a proprietary binary representation. The XML/GML representation is preferred as permitting easier interchange between different commercial entities, while the binary representation is preferred as more compact and readily transferable to a client system **18**, **20**. In both cases, the source overlay data **34** is pre-processed to contain the annotation data preferably in a resolution independent form associated with a display coordinate specification relative to the source image data **32**. The XML, GML or binary overlay data may be compressed prior to storage on the network server **12**, **22**.

The preferred architecture **40** of a client system **18**, **20**, for purposes of implementing the present invention, is shown in FIG. **3**. The architecture **40** is preferably implemented by software plug-in or application executed by the client system **18**, **20** and that utilizes basic software and hardware services provided by the client system **18**, **20**. A parcel request client **42** preferably implements an HTML client that supports HTML-based interactions with the server **12**, **22** using the underlying network protocol stack and hardware network interface provided by the client systems **18**, **20**. A central parcel processing control block **44** preferably implements the client process and control algorithms. The control block **44** directs the transfer of received image parcels and XML/GML/binary overlay data to a local parcel data store **46**. Local parcel data store **46** may also act for example as local cache weather the entire data or part of it is in dynamic and/or static cache. Preferably image data parcels are stored in conventional quad-tree data structures, where tree nodes of depth D correspond to the stored image parcels of a derivative image of resolution KD. The XML/GML/binary overlay data is preferably stored as a data object that can be subsequently read by an XML/GML/binary parser implemented as part of the control block **44**.

The control block **44** is also responsible for decompressing and directing the rendering of image parcels to a local display by a rendering engine **48**. Preferably, the rendering engine **48** writes to the video memory of the underlying client display hardware relying on only generic graphics acceleration hardware capabilities and may take advantage of more advanced graphics acceleration hardware when available in the client system **18**, **20**. In general, the relied on capabilities include bit-bit and related bit-oriented functions that are readily supported by current conventional display controller hardware. The rendering engine **48** is optimized to perform image parcel texture mapping without reliance on complex floating point operations, permitting even relatively simple processors to efficiently execute the rendering engine **48**. The rendering engine **48** may take advantage of floating point operations when available in the client system **18**, **20**.

Changes in the viewing frustum are determined from user input navigation commands by a frustum navigation block **50**. In the preferred embodiments of the present invention, the input navigation controls are modeled for three-dimensional fly-over navigation of the displayed image. The navigation controls support point-of-view rotation, translation, attitude, and altitude over the displayed image. The effective change in viewing frustum as determined by the frustum navigation block **50** is provided to the control block **44**.

The control block **44**, based in part on changes in the viewing frustum, determines the ordered priority of image parcels to be requested from the server **12**, **22** to support the progressive rendering of the displayed image. The image parcel requests are placed in a request queue **52** for issuance by the parcel request client **42**. Preferably, the pending requests are issued in priority order, thereby dynamically reflecting changes in the viewing frustum with minimum latency.

In various implementations of the parcel processing, each data parcel is independently processable by the client system **18**, **20**, which is enabled by the selection and server-side processing used to prepare a parcel for transmission, thus providing for on-demand real-time parcel processing and creation on the server side for streaming based on the client request and not only for pre-processed parcel creation for retrieval for streaming from the server. Thus, the system can use both pre-processed parcels on the server and on-demand real-time creation of such parcels on the server side for streaming to the client.

An optimal image parcel data flow **60**, as configured for use in the preferred embodiments of the present invention, is shown in FIG. **4**. Preferably, the TCP/IP network protocol is used to deliver image parcels to the clients **18**, **20**. For the preferred embodiments, where network bandwidth is limited or very limited, entire image parcels are preferably delivered in corresponding data packets. This preference maximizes data delivery while avoiding the substantial latency and processing overhead of managing image parcel data split over multiple network packets. Thus, a 2K byte compressed image parcel **62** is delivered as the data payload of a TCP/IP packet **64**. Uncompressed, the 8K byte image parcel **62** is recognized as part of the present invention as being within the nominally smallest LI data cache **66** size of conventional microprocessors **68**. By ensuring that an uncompressed image parcel fits within the LI cache, the texture map rendering algorithm can execute with minimum memory management overhead, thus optimally utilizing the processing capability of the microprocessor **68**. Additionally, the writing of video data as a product of the rendering algorithm is uniform, thereby improving the apparent video stability of the display to the user.

The client architecture **40** preferably executes in multiple process threads, with additional threads being utilized for individual network data request transactions. As shown in FIG. **5**, an image parcel management process **80** implements a loop that determines image parcels subject to update **82** and creates corresponding image parcel download requests **84**. Navigation events that alter the viewing frustum are considered in part to determine the current field of view. The quad tree data structures are examined **86** to identify viewable image parcels of higher resolution than currently available in the parcel data store **46**.

A pool of image request threads is preferably utilized to manage the image parcel download operations. In the preferred embodiments of the present invention, a pool of four network request threads is utilized. The number of pool threads is determined as a balance between the available system resources and the network response latency, given the available bandwidth of the network connection. Empirically, for many wireless devices, four concurrent threads are able to support a relatively continuous delivery of image data parcels to the client **20** for display processing. As image parcels are progressively identified for download, a free request thread is employed to issue **88** a corresponding network request to the server **12**, **22**. When a network response is received, the corresponding thread recovers **90** the image parcel data. The received image parcel is then stored **92** in a corresponding quad-tree data structure node.

For small clients **20**, the available memory for the parcel data store **46** is generally quite restricted. In order to make optimal use of the available memory, only currently viewable image parcels are subject to download. Where the size of the parcel data store **46** is not so restricted, this constraint can be relaxed. In either case, a memory management process **94** runs to monitor use of the parcel data store **46** and selectively remove image parcels to free memory for newly requested image parcels. Preferably, the memory management process **94** operates to preferentially remove image parcels that are the furthest from the current viewing frustum and that have the highest data structure depth. Preferably child node image parcels are always removed before a parent node parcel is removed.

A preferred network request management process **100** is shown in FIG. **6**. The process **100** waits **102** on the existence of a download request in the priority request queue **52**. The process **100** then waits on a network request pool thread to become free **104**. When a network request thread becomes available, the process **100** examines **106** all of the pending requests in the priority request queue **52** and selects **108** the request with the highest assigned priority. Thus, sequentially enqueued requests can be selectively issued out of order based on an independently assigned request priority. The request is then issued **110** and the request management process **100** leaves the request thread waiting on a network response.

FIG. **7** presents a preferred display management process **120**. Event driven user navigation information is evaluated **122** to determine a current viewing frustum location and orientation within a three-dimensional space relative to the displayed image. An algorithmic priority selection **124** of a next image parcel to render is then performed. The selected image parcel is then rendered **126** to the display memory **70**. The rendering operation preferably performs a texture map transform of the parcel data corresponding to the current viewing frustum location and orientation. The overlay data is then parsed or is pre-parsed to determine **128** whether the image coordinates of any overlay annotation correspond to the current image parcel location. If the coordinates match, the overlay annotation is rendered **130** to the video display memory **70**. The process **120** then continues with the next selection **124** of an image parcel to render, subject to any change in the viewing frustum location and orientation.

A preferred implementation of the selection **124** and rendering **126** of image parcels in accordance with the present invention is detailed in FIGS. **8** through **10**. Referring first to FIG. **8**, any outstanding requests in the priority request queue **52** are preferably cleared **142** in response to a change in the viewing frustum location and orientation. The effective altitude of the viewing frustum and or the resolution of the client display are then used as a basis for determining an optimal level of detail L that will be displayed. The detail level L value operates as a floor defining the maximum resolution K.sub.L of image data that can be effectively viewed on the client display given the location and or orientation of the viewing frustum. Constraining image parcel requests to the resolution range K.sub.N to K.sub.L, where K.sub.N is the lowest resolution derivative image stored by the network server **12**, **22**, prevents the download and processing of image parcels that cannot provide any perceptible improvement in the displayed image.

As part of the recursive evaluation of the optimal level of detail L, the image display space is progressively split **146** by four to one reductions into polygons. The quad-tree data structures holding existing image parcel data in the parcel data store **46** are concurrently traced **148** to establish a correspondence with the polygon map. Where the trace of a quad-tree data structure completes **150** to a node index of L for a polygon P, the node corresponding image parcel is associated with polygon P. The polygon P will not be further subdivided and no higher resolution image parcels will be requested for any portion of the image within the area represented by polygon P. Where the trace reaches a maximum node index of D for a polygon P' **152**, where N.ltoreq.D<L and N is the index of the lowest resolution derivative image stored by the network server **12**, **22**, the image parcel associated with the node is associated with the polygon P'. This polygon P' will be subject to further subdivision and progressive requests for image parcels of higher resolution up to the detail level L.

Referring now to FIG. **9**, a display image is then rendered **160** beginning with the maximum depth polygons previously found. Iterating over the set of maximum depth polygons, any polygons outside of the viewing frustum are skipped **162**. Polygons that are at least partially visible are clipped to the applicable bounds of the viewing frustum **164**. The polygon corresponding image parcel data is then texture mapped **166** into the polygon corresponding coordinates of the video memory **70**. If the node index depth of the rendered image parcel is at least equal to the prior determined optimal detail level L **168**, the iteration over the polygons P continues.

Where the node index depth is less than the optimal detail level L **170**, the polygon P' is subdivided into four polygons and correspondingly represented by the creation of four child nodes within the associated quad-tree data structure **172**. Four image parcel download requests are then created **174**.

The download priority associated with each request is determined **176** by execution of a function S that operates on a 2D polygon argument P and returns a real number representing the request priority. The function argument P is a list of real (x, y) coordinates of the vertices of the current polygon in screen coordinates after being clipped to fit within the current viewing frustum. That is, the function S works over general polygons in a two-dimensional space, whose vertices are specified by the series $\{(x(1), y(1)), (x(2), y(2)), \ldots, (x(n), y(n))\}$. The argument P vertices sent to S represent the position of the vertices composing each of the polygons, after being clipping to the viewing frustum, viewable within the display space having the fixed resolution [xRes, yRes]. Thus, the clipped polygons are all within the rectangle [O, xRes]× [O, yRes].

In execution of the function S, each of the P coordinates is first transformed by linear mapping of the screen coordinate space to the square [−1,1].times. [−1,1] by the operation x(i):=(x(i)−xRes/2)/(xRes/2); y(i)=(y(i)−yRes/2)/(yRes/2). The x and y coordinate values of each vertex (x(i),y(i)) for i=1 to n) are then transformed by the function T(a)=sgn(a)*pow (.vertline.a.vertline., d), where the control parameter d is a constant in the range (0,1], or equivalently the interval 0<dltoreq.1. The function S then returns a real value that is equal to the area covered by the argument polygon P vertices subject to the applied coordinate transformation. Thus, the accumulated priority for any image parcel pending download is the sum of the values of returned by the function S for each of the viewable polygons that require some part of the image parcel as the source data for texture map rendering of the polygon. The priority operation of the request queue **52** is such that download requests will be issued preferentially for image parcels with the largest priority value.

In accordance with the preferred embodiments of the present invention, the value of the control parameter d can be adjusted to ultimately affect the behavior of the function S in determining the download request priority. In general, image parcels with lower resolution levels will accumulate greater priority values due to the larger number of polygons that may use a given low resolution image parcel as a rendering data source. Such lower resolution image parcels are therefore more likely to be preferentially downloaded. In accordance with the present invention, this generally assures that a complete image of at least low resolution will be available for rendering.

The control parameter d, as applied in execution of the function S, well as the area distortion produced by the projection transform also influences the value returned by the function S such that relatively higher-resolution image parcels near the image view point will occasionally achieve a higher priority than relatively remote and partially viewed image parcels of lower resolution. Using values smaller than 1 for the control parameter d results in requests with a higher priority for parcels covering areas near the focal point of the viewer, which is presumed to be the center point of the display space, relative to requests for parcels further from the center point in absolute terms and of the same resolution depth D. Thus, in accordance with the present invention, the priority assigned to image parcel requests effectively influences the order of requests based on the relative contribution of the image parcel data to the total display quality of the image. Empirically, a value of 0.35 for the control parameter d for small screen devices, such as PDAs and webphones has been found to produce desirable results.

The computed priorities of each of the four newly created image parcel requests are then assigned **178** and the requests are enqueued in the priority request queue **52**. The next polygon P is then considered in the loop of the image parcel rendering process **160**.

The preferred algorithm **180** for determining the detail level L value for a given viewing frustum is shown in FIG. **10**. In accordance with the present invention, the optimal detail level L is effectively the limit at which the resolution of image parcel data functionally exceeds the resolution of the client display. Preferably, to determine the optimal detail level L, the viewpoint or camera position of the viewing frustum is determined **182** relative to the displayed image. A nearest polygon P of depth D is then determined **184** from the effective altitude and attitude of the viewpoint. The nearest point A of the polygon P is then determined **186**. The point A may be within the interior or an edge of the polygon P, though most likely be located at a vertex of the polygon P.

The optimum level of detail L at point A is then computed **188** as the base-4 logarithm of the number of pixels on the screen that would be covered by a single pixel from an image parcel of the lowest resolution K.sub.-N image, which is the quad-tree root image and corresponds to an image area covering the entire image map. The point A optimal detail level L is preferably computed analytically from the local value of the Jacobian of the projective transform used to transform the three dimensional image coordinate space to screen coordinates, evaluated at the point A.

Where the depth D of the polygon P is greater than the depth of the computed optimal level of detail L, the detail level L is taken as the optimal detail level L **190**. Thus, through the process **140**, an image parcel or corresponding section of the closest resolution image parcel associated with a parent node in the quad-tree data structure relative to the depth level L will be used as the texture for rendering the polygon P. Conversely, if the depth D is less than that of the optimal detail level L, the polygon P is effectively split into quadrants and the optimal level of detail is reevaluated. The process **180** thus continues iteratively until the optimal detail level L is found.

Thus, a system and methods of optimally presenting image data on client systems with potentially limited processing performance, resources, and communications bandwidth have been described. While the present invention has been described particularly with reference to the communications and display of geographic image data, the present invention is equally applicable to the efficient communications and display of other high resolution information.

In the process implemented by the system described above, data parcels may be selected for sequential transmission based on a prioritization of the importance of the data contained. The criteria for the importance of a particular data parcel may be defined as suitable for particular applications and may directly relate to the presentation of image quality, provision of a textual overlay of a low-quality image to quickly provide a navigational orientation, or the addition of topography information at a rate or timing different from the rate of image quality improvement. Thus, image data layers reflecting navigational cues, text overlays, and topography can be composed into data packets for transmission subject to prioritizations set by the server alone and not based on the client system and interactively influenced by the actions and commands provided by the user of the client system. However this also may be influenced based on the nature and type of the client system, and interactively influenced by the actions and commands provided by the user of the client system (FIG. **5**).

In view of the above description of the preferred embodiments of the present invention, many modifications and variations of the disclosed embodiments will be readily appreciated by those of skill in the art. It is therefore to be understood that, within the scope of the appended claims, the invention may be practiced otherwise than as specifically described above.

The invention claimed is:

1. A method of retrieving images over a network communication channel for display on a user computing device, the method comprising steps of:

issuing a first request from the user computing device to one or more servers, over one or more network communication channels, the first request being for a first update data parcel corresponding to a first derivative image of a predetermined image, the predetermined image corresponding to source image data, the first update data parcel uniquely forming a first discrete portion of the predetermined image, wherein the first update data parcel is selected based on a first user-controlled image viewpoint on the user computing device relative to the predetermined image;

receiving the first update data parcel at the user computing device from the one or more servers over the one or more network communication channels, the step of receiving the first update data parcel being performed after the step of issuing the first request;

displaying the first discrete portion on the user computing device using the first update data parcel, the step of displaying the first discrete portion being performed after the step of receiving the first update data parcel;

issuing a second request from the user computing device to the one or more servers, over the one or more network communication channels, the second request being for a second update data parcel corresponding to a second derivative image of the predetermined image, the second update data parcel uniquely forming a second discrete portion of the predetermined image, wherein the second update data parcel is selected based on a second user-controlled image viewpoint on the user computing device relative to the predetermined image, the second user-controlled image viewpoint being different from the first user-controlled image viewpoint;

receiving the second update data parcel at the user computing device from the one or more servers over the one or more network communication channels, the step of receiving the second update data parcel being performed after the step of issuing the second request;

displaying the second discrete portion on the user computing device using the second update data parcel, the step of displaying the second discrete portion being performed after the step of receiving the second update data parcel;

wherein:

a series of K1-N derivative images of progressively lower image resolution comprises the first derivative image and the second derivative image, the series of K1-N of derivative images resulting from processing the source image data, series image K0 being subdivided into a regular array wherein each resulting image parcel of the array has a predetermined pixel resolution and a predetermined color or bit per pixel depth, resolution of the series K1-N of derivative images being related to resolution of the source image data or predecessor image in the series by a factor of two, and the array subdivision being related by a factor of two.

2. A method as in claim **1**, further comprising:

determining the first user-controlled image viewpoint based at least in part on first navigational input of the user computing device; and

preparing the first request by a processing control block of the user computing device based at least in part on the first user-controlled image viewpoint.

3. A method as in claim **2**, wherein the step of preparing the first request is performed based at least in part on altitude and attitude of the first viewpoint relative to the predetermined image.

4. A method as in claim **2**, further comprising:

determining the second user-controlled image viewpoint based at least in part on second navigational input of the user computing device; and

preparing the second request by a processing control block of the user computing device based at least in part on the second user-controlled image viewpoint.

5. A method as in claim **4**, wherein:

the step of preparing the first request is performed based at least in part on three-dimensional altitude and attitude of the first viewpoint relative to the predetermined image; and

the step of preparing the second request is performed based at least in part on three-dimensional altitude and attitude of the second viewpoint relative to the predetermined image.

6. A method as in claim **5**, wherein the predetermined image is an image of a geographic area.

7. A method as in claim **5**, wherein:

the first navigational input comprises first lateral x dimension position data, first lateral y dimension position data, first z height dimension position data, and first rotational position data; and

the second navigational input comprises second lateral x dimension position data, second lateral y dimension position data, second z height dimension position data, and second rotational position data.

8. A method as in claim **5**, wherein:

the first navigational input comprises first three-dimensional coordinate position data and first rotational position data; and

the second navigational input comprises second three-dimensional position data and second rotational position data.

9. A method as in claim **1**, wherein:

the first derivative image includes the second derivative image;

the second derivative image has a higher level of detail than the first derivative image; and the step of issuing the first request is performed before the step of issuing the second request.

10. A method as in claim **1**, wherein:

the second derivative image includes the first derivative image;

the second derivative image has a lower level of detail than the first derivative image; and

the step of issuing the first request is performed before the step of issuing the second request.

11. A method as in claim **1**, wherein the first derivative image does not include the second derivative image, and the second derivative image does not include the first derivative image.

12. A method as in claim **1**, wherein the first update data parcel comprises first overlay data for the first derivative image.

13. A method as in claim **12**, wherein the first overlay data comprises first text annotation relating to at least one of: one or more street names, one or more building names, and one or more landmarks.

14. A method as in claim **12**, wherein the first overlay data comprises graphic data representing a three-dimensional object.

15. A method as in claim **12**, wherein the first overlay data comprises graphics data describing at least one object in three dimensions.

16. A method as in claim **12**, wherein the first overlay data comprises one or more graphical icons.

17. A method as in claim **12**, wherein the second update data parcel comprises second overlay data for the second derivative image.

18. A method as in claim **17**, wherein the first overlay data and the second overlay data are in a resolution-independent format.

19. A method as in claim **12**, wherein the first overlay data comprises first text annotation relating to at least one of: one or more street names, one or more building names, and one or more landmarks.

20. A method as in claim **1**, further comprising a step for determining priority of the first request and the second request.

21. A method as in claim **1**, wherein:

the steps of issuing the first request and receiving the first update data parcel are part of a first thread;

the steps of issuing the second request and receiving the second update data parcel are part of a second thread; and

the first thread and the second thread are executed at least in part concurrently.

22. A method as in claim **1**, further comprising:

issuing a third request from the user computing device to the one or more servers, over the one or more network communication channels, the third request being for a third update data parcel corresponding to a third derivative image of the predetermined image, the third update data parcel uniquely forming a third discrete portion of the predetermined image;

receiving the third update data parcel at the user computing device from the one or more servers over the one or more network communication channels;

issuing a fourth request from the user computing device to the one or more servers, over the one or more network communication channels, the fourth request being for a fourth update data parcel corresponding to a fourth derivative image of the predetermined image, the fourth update data parcel uniquely forming a fourth discrete portion of the predetermined image; and

receiving the fourth update data parcel at the user computing device from the one or more servers over the one or more network communication channels;

wherein:

the steps of issuing the first request and receiving the first update data parcel are part of a first thread;

the steps of issuing the second request and receiving the second update data parcel are part of a second thread;

the steps of issuing the third request and receiving the third update data parcel are part of a third thread;

the steps of issuing the fourth request and receiving the fourth update data parcel are part of a fourth thread; and

the first thread, the second thread, the third thread, and the fourth thread are executed at least in part concurrently.

23. A method as in claim **1**, wherein the user computing device is a mobile device.

24. A method as in claim **1**, wherein the one or more servers comprise at least two servers.

25. A method as in claim **1**, wherein each image parcel is of a fixed byte size.

* * * * *