Kent Walker (State Bar No. 173700)
Lewis Kohn & Walker LLP
15030 Avenue of Science, Suite 201
San Diego CA 92128
Telephone: (858) 436-1330
Fax: (858) 436-1349

(Additional Counsel Identified On Signature Page)
Attorneys for Plaintiff CYWEE GROUP LTD.

IN THE UNITED STATES DISTRICT COURT
FOR THE SOUTHERN DISTRICT OF CALIFORNIA

CYWEE GROUP LTD.,
Plaintiff,

LG ELECTRONICS, INC.,
LG ELECTRONICS U.S.A., INC.,
AND LG ELECTRONICS
MOBILECOMM U.S.A., INC.,
Defendants.

CASE NO. 3:17-cv-01102

CYWEE'S FIRST AMENDED COMPLAINT FOR PATENT INFRINGEMENT

DEMAND FOR JURY TRIAL

1. Plaintiff CyWee Group Ltd. ("Plaintiff" or "CyWee") by and through its undersigned counsel, files this Amended Complaint against Defendants LG Electronics, Inc., LG Electronics U.S.A., Inc. and LG Electronics MobileComm U.S.A., Inc. ("Defendants" or "LG) as follows:

THE PARTIES

2. CyWee is a corporation existing under the laws of the British Virgin Islands with a principal place of business at 3F, No.28, Lane 128, Jing Ye 1st Road, Taipei, Taiwan 10462.

3. CyWee is a world-leading technology company that focuses on building products and providing services for consumers and businesses. CyWee has one of the most significant patent portfolios in the industry, and is a market leader in its core development areas of motion processing, wireless high definition video delivery, and facial tracking technology.

4. On information and belief, Defendant LG Electronics, Inc. ("LGE") is a company incorporated in South Korea located at LG Twin Tower, 128 Yeoui-daero, Yeongdeungpo-gu, Seoul, 150-721, South Korea.

5. On information and belief, Defendant LG Electronics U.S.A., Inc. ("LGEUSA") is a Delaware corporation with its principal place of business at 920 Sylvan Avenue, Englewood Cliffs, New Jersey 07632. LGEUSA is a wholly owned subsidiary of LGE. Dkt. No. 37. On information and belief, LGEUSA may be served via its registered agent for service of process: Lawyers Incorporating Service, 2710 Gateway Oaks Drive, Suite 150N, Sacramento CA 95833.

6. On information and belief, Defendant LG Electronics MobileComm U.S.A., Inc. ("LGEMU") is a California corporation with its principal place of business at 10101 Old Grove Road, San Diego, California 92131. LGEMU is a wholly-owned subsidiary of LGEUSA. Dkt. No. 37. On information and belief, LGEMU may be served via its registered agent for service of process: CSC – Lawyers

Incorporating Service, 2710 Gateway Oaks Drive, Suite 150N, Sacramento, CA 95833.

7. Defendants LGE, LGEUSA, and LGEMU are collectively referred to as "Defendants" or "LG." LG is doing business in the United States and, more particularly, in the State of California and the Southern District of California, by designing, marketing, making, using, selling, importing, and/or offering for sale products that infringe the patent claims involved in this action or by transacting other business in this District.

JURISDICTION AND VENUE

8. This action arises under the patent laws of the United States, 35 U.S.C. § 1 *et seq*. This Court has subject matter jurisdiction pursuant to 28 U.S.C. §§ 1331 and 1338(a).

9. This Court has personal jurisdiction over each Defendant. Each Defendant has conducted and does conduct business within the State of California. Each Defendant has purposefully and voluntarily availed itself of the privileges of conducting business in the United States, State of California, and in the Southern District of California by continuously and systematically placing goods into the stream of commerce through an established distribution channel with the expectation that they will be purchased by consumers in the Southern District of California. Plaintiff's cause of action arises directly from Defendants' business contacts and other activities in the State of California and the Southern District of California.

10. Upon information and belief, each Defendant has committed acts of infringement in this District giving rise to this action and does business in this District, including making sales and/or providing service and support for their respective customers in this District. Defendants purposefully and voluntarily sold one or more of their infringing products with the expectation that they will be purchased by consumers in this District. These infringing products have been and continue to be

purchased by consumers in this District. Defendants have committed acts of patent infringement within the United States and, state of California, and the Southern District of California.

11. Venue is proper as to LGEMU under 28 U.S.C. § 1400(b) in that LGEMU is incorporated in California and, therefore, resides in this District. *TC Heartland LLC v. Kraft Foods Grp. Brands LLC*, 137 S. Ct. 1514, 1521 (2017).

12. Upon information and belief, LGEMU is an agent of LGEUSA and is held out to the public as such. *See* http://www.lg.com/us/careers (last visited May 30, 2017) (referring to "LG MobileComm USA" as a "LG Mobile Unit division"). Upon information and belief, LGEMU does not have its own website but is listed as the "Media Contact" for "Mobile Phones" on the LG.com United States website. *See* http://www.lg.com/us/press-media/media-contacts (last visited May 30, 2017).

13. Further, upon information and belief, LGEMU operates under the "LG" trademark; offers, sells, services, and/or distributes only LG products; and coordinates its policies and operations with those of LGEUSA to benefit and primarily serve the interests of LGEUSA and LGEUSA's parent corporation.

14. Further, upon information and belief, support materials and documentation provided to consumers with the mobile products offered by Defendants do not delineate between LGE, LGEUSA, and LGEMU. By way of example, the User Guide for the LG G6 refers generally to "LG" without any distinction as to LGE, LGEUSA, or LGEMU, including in the limited warranty provided therein. Upon information and belief, for consumers of the products accused in this Complaint, there is no substantive distinction between LGEMU and either LGEUSA or LGE.

15. Based on the foregoing, venue is proper as to LGEUSA under 28 U.S.C. § 1400(b) in that, upon information and belief, LGEUSA has a regular and established place of business in this District—namely, the place of business of its

subsidiary/agent, LGEMU—and has committed acts of infringement herein. *See Appleton v. Ronson Serv. of Ill., Inc.*, 297 F.Supp. 868, 869 (N.D. Ill. 1968) ("formal corporate separateness can be disregarded for the purpose of establishing venue . . . where a number of factors . . . in the aggregate reveal a mere cloak for the relationship of agency." (citing *Leach Co. v. Gen. Sani-Can Mfg. Corp.*, 393 F.2d 183 (7th Cir. 1968)); *see also Stanley Works v. Globemaster, Inc.*, 400 F.Supp. 1325, 1331-32 (D. Mass. 1975).

16. Venue is proper as to LGE under 28 U.S.C. § 1391(c)(3) in that it is not a resident of the United States and may, therefore, be sued in any judicial district. *Brunette Mach. Works, Ltd. v. Kockum Indus., Inc.*, 406 U.S. 706, 714 (1972). Venue is further proper as to LGE under 28 U.S.C. § 1400(b) in that, upon information and belief, LGE has a regular and established place of business in this District—namely, the place of business of its subsidiaries/agents, LGEUSA and LGEMU—and has committed acts of infringement herein.

WILLFUL INFRINGEMENT

17. LG's infringement of the patents-in-suit has been and continues to be willful. LG has had knowledge of and notice of both patents-in-suit and its infringement of those patents as a result of confidential pre-suit licensing discussions. LG also has knowledge and notice of its infringement of the patents-in-suit as a result of the complaints filed in this case, which include two claim charts illustrating LG's infringement of the patents-in-suit. LG's infringement of the patents-in-suit has been and continues to be willful and deliberate.

BACKGROUND

Patentee And The Asserted Patents.

18. ITRI is a Taiwanese government and industry funded research and development center. In 2007, CyWee, which was started at ITRI, was formed. Its goal was to provide innovative motion-sensing technologies, such as those claimed in the patents-in-suit. Dr. Shun-Nan Liu and Chin-Lung Li, two of the inventors of the Patents, came to CyWee from ITRI. The third inventor, Zhou "Joe" Ye joined CyWee as its President and CEO from private industry.

19. The inventors, Zhou Ye, Chin-Lung Li, and Shun-Nan Liou conceived of the claims of the patents-in-suit—U.S. Patent No. 8,441,438 (the "'438 patent") and U.S. Patent No. 8,552,978 (the "'978 patent")—at CyWee Group Ltd., located at 3F, No. 28, Lane 128, Jing Ye Road, Taipei.

20. Several claims of the patents-in-suit are entitled to a priority date of at least January 6, 2010 based on U.S. Provisional Application Serial No. 61/292,558, filed January 6, 2010 ("Provisional Application").

21. Before May 22, 2009, CyWee began working on the "JIL Game Phone Project" or "JIL Phone." Before July 29, 2009, CyWee developed a solution for the JIL Phone that practiced several claims of the '438 patent. Those claims were diligently and constructively reduced to practice thereafter through the filing of the Provisional Application, and were diligently and actually reduced to practice as discussed below. Accordingly, CyWee is entitled to a priority date of at least July 29, 2009 for several claims of the '438 patent.

22. The JIL Phone was reduced to practice by at least September 25, 2009. The JIL Phone practiced several claims of both patents-in-suit. Accordingly, CyWee is entitled to a priority date of at least September 25, 2009 for several claims of the patents-in-suit.

Background Of The Technology.

23. The ’438 patent and ’978 patent are each directed to devices and methods for tracking the motion of a portable electronic device in 3D space and compensating for accumulated errors to map the 3D movements of the device onto a display frame (’438 patent) or transform the 3D movements for a display, such as a 2D display on a computer or handheld device (’978 patent). ’438 patent 1:17-52, 3:52-57; ’978 patent 1:22-27, 7:5-18; Exhibit C, Declaration of Nicholas Gans, Ph.D. (“Gans Decl.”) ¶ 8.[1] At a high level, the patented inventions teach how to determine a device’s current orientation based on motion data detected by its motion sensors, such as an accelerometer, gyroscope, and magnetometer. ’438 patent 4:6-30; ’978 patent 4:15-44; Gans Decl. ¶ 8. The ’438 and ’978 patents describe portable electronic devices or pointing devices such as smartphones and navigation equipment. ’978 patent 22:34-40, Fig. 6; ’438 patent 4:6-30, Fig. 6; Gans. Decl. ¶ 8.

24. There are different types of motion sensors, including accelerometers, gyroscopes, and magnetometers. Gans Decl. ¶ 9. Accelerometers measure accelerations. *Id.* For example, airbags use accelerometers, such that the airbag is triggered based on sudden deceleration. Accelerometers can also measure forces due to gravity. *Id.* Gyroscopes measure rotation rates or angular velocities. Magnetometers measure magnetism, including the strength of a magnetic field along a particular direction. *Id.* Each type of motion sensor is subject to inaccuracies. *Id.* For example, a gyroscope sensor has a small, added offset or bias. *Id.* This bias will accumulate over time and lead to large drift error. *Id.* Similarly, magnetometers are subject to interference from natural and manmade sources (*e.g.*, power electronics). *Id.* Additionally, errors can accumulate over time. *Id.* These sensors typically take measurements along a single direction. *Id.* To accurately measure motions along an

---

[1] The declaration of Dr. Nicholas Gans is incorporated by reference herein.

arbitrary axis, three like sensors are grouped together and aligned at right angles. Such a sensor set is generally referred to as a 3-axis sensor. *Id.*

25. Orientation information returned by the claimed inventions of the ’438 and ’978 patents has many uses, particularly for mobile cellular devices, such as navigation, gaming, and augmented/virtual reality applications. Gans Decl. ¶ 12. Navigation applications can use orientation information to determine the heading of the phone, indicate what direction the user is facing, and automatically orient the map to align with the cardinal directions. *Id.* Increasing numbers of games and other applications use the motion of the phone to input commands, such as tilting the mobile device like a steering wheel. *Id.* Augmented and virtual reality applications rely on accurate estimation of the device orientation in order to render graphics and images at the proper locations on the screen. *Id.*

26. Prior to 2010, motion sensors had limited applicability to portable electronic devices due to a variety of technological hurdles. Gans Decl. ¶ 13. For example, different types of acceleration (*e.g.*, linear, centrifugal, gravitational) could not be readily distinguished from one another, and rapid, dynamic, and unexpected movements caused significant errors and inaccuracies. *Id.* These difficulties were compounded by the miniaturization of the sensors necessary to incorporate them in portable electronic devices. *Id.* With the development of micro-electromechanical systems, or “MEMS,” miniaturized motion sensors could be manufactured and incorporated on a semiconductor chip, but such MEMS sensors had significant limitations. *Id.*

27. For example, it is impossible for MEMS accelerometers to distinguish different types of acceleration (*e.g.*, linear, centrifugal, gravitational). Gans Decl. ¶ 14. When a MEMS accelerometer is used to estimate orientation, it must measure force along the direction of gravity (*i.e.*, down), but that gravitational measurement can be “interfused” with other accelerations and forces (*e.g.*, vibration or movement

by the person holding the device). *Id.* Thus, non-gravitational accelerations and forces must be estimated and subtracted from the MEMS accelerometer measurement to yield an accurate result. *Id.* A MEMS gyroscope is prone to drift, which will accumulate increasing errors over time if not corrected by another sensor or recalibrated. *Id.* A MEMS magnetometer is highly sensitive to not only the earth's magnetic fields, but other sources of magnetism (*e.g.*, power lines and transformers) and can thereby suffer inaccuracies from environmental sources of interference that vary both in existence and intensity from location to location. *Id.*

28. Additionally, orientation cannot be accurately calculated using only one type of MEMS sensor. Gans Decl. ¶ 15. For example, if only a 3-axis MEMS accelerometer is used to measure orientation, pitch and yaw can be measured, but not roll. *Id.* If only a MEMS gyroscope is used to measure angular velocity, only relative changes in orientation can be measured, not absolute orientation. *Id.*

29. Without orientation information, mobile device apps would be limited to very static operation. Gans Decl. ¶ 16. This was the scenario with initial smart phones and other mobile devices. *Id.* Navigation aids could render a map and indicate the location of the device using GPS. *Id.* However, these maps would orient with North on the map pointing to the top of the screen. *Id.* The user could rotate the map using touch commands, but the map would not rotate automatically as the user turned. *Id.* Nor could the device indicate what direction the device was facing. *Id.*

30. Many games use motion of the device to control the game. Gans Decl. ¶ 17. A common control scheme, especially for driving and piloting games, is to have the user rotate the device, such as a phone or game controller, like a steering wheel to indicate the direction the vehicle should move. *Id.* Some puzzle games also use motions to cause elements of the game to move. *Id.* As discussed previously, accelerometers measure acceleration, which is a very noisy signal. *Id.* Acceleration is the derivative of velocity, which is the derivative of position. *Id.* Small magnitude

noise can have large derivatives, which means that small levels of noise from vibration or electrical fluctuations will be magnified at the acceleration level. *Id.* Even a stationary device will have notable noise measured by an accelerometer. *Id.* A moving device will only amplify this noise. *Id.* Since accelerometers measure linear and centripetal accelerations as well as the acceleration of gravity, orientation estimates on a moving device will not be accurate. *Id.*

31. If only an accelerometer is used, a coarse estimate of the device orientation can be obtained by averaging or numerically filtering the results. Gans Decl. ¶ 18. Essentially, the device can determine if it is tilted left or right, up or down, but the exact angle cannot be estimated accurately while in motion. *Id.* This is suitable for games to move a character or steer a vehicle in a particular direction, but generally cannot utilize the magnitude of tilt to move at corresponding faster or slower speeds. *Id.*

32. Movement on a display can, of course, be controlled by means other than a portable electronic device with orientation sensors. Gans Decl. ¶ 19. For example, games could be controlled using traditional "joystick" type inputs. *Id.* For smart phones with touch screens, commands are given by having the user touch specific parts of the screen. *Id.*

33. For other current applications, portable electronic devices with orientation sensors are more crucial. Gans Decl. ¶ 20. Augmented reality (AR) and virtual reality (VR) are new and growing classes of applications for smart phones and mobile devices. *Id.* In AR, the device camera provides live video feed to the screen, and the application overlays generate graphics onto the screen at specific locations. *Id.* AR navigation apps can draw signs or labels to indicate what specific places or objects are, or can render arrows or other indicators. *Id.* AR games and teaching applications can label objects or draw characters or items such that they appear as if they are in the real world seen in the video. *Id.* Virtual reality is similar but does not

use the camera, rather it completely renders an artificial 3D environment on the screen. *Id.* VR most often requires a head set such that the user only sees the screen. *Id.* Mobile devices and smart phones used for VR generally split the screen and display to two side-by-side images of the rendered environment that are slightly offset to simulate a left and right eye. *Id.* The device then sits in a headset with lenses such that the user has each eye see only one of the split-screen images and has a sense of stereo (3D) vision. *Id.*

34. Without orientation sensing, AR and VR applications cannot work. Gans Decl. ¶ 21. The system will have no ability to understand the orientation of the device and know where to draw objects and/or the scene. *Id.* The rough orientation estimate provided by an accelerometer (ideally with a magnetometer) will not be sufficient to track during typical head motions. *Id.* It has been demonstrated that VR applications that use an accelerometer often cause motion sickness, as the rendered images do track with the head motions. *Id.* An AR application with the use of a gyroscope and fusion algorithm will not render objects at the correct locations, and may obscure the view rather than provide helpful information. *Id.*

35. There are ways to estimate orientation other than the approaches presented in the ’438 patent or ’978 patent, which involve algorithms that filter and fuse measurements from inertial and magnetic sensors. Gans Decl. ¶ 22. Most such methods are based on cameras and computer vision algorithms. *Id.* However, the limitations of these methods render them unusable for portable electronic devices. *Id.* For example, there are a variety of motion capture systems that use cameras arrayed around an environment. Markers (*e.g.*, reflective balls) can be placed on objects, and the cameras can locate the markers, often to sub-mm accuracy. *Id.* If an object has three or more markers on it, the orientation of the object can be determined with sub-degree accuracy. *Id.* This method is very accurate, but quite expensive (often about $100,000). *Id.* The cameras are fixed in place, and the estimation can only work within

a small space (a box of dimensions on the order of tens of meters). *Id.* This is not suitable for the vast majority of mobile device users or applications. *Id.*

36. A camera on a portable electronic device, such as a smart phone, can be used to estimate orientation of the phone. Gans Decl. ¶ 23. One class of approaches to this problem uses special patterns or markers in the environment. *Id.* These often have the appearance of a QR code or 2D UPC. *Id.* Taking a picture of the pattern, computer vision algorithms can determine the position and orientation of the camera with respect to the marker. *Id.* AR applications have placed the patterns on specific objects or consumer products so the device can render images and graphics with respect to the pattern. *Id.* AR games have included patterned mats that are placed on a table or other flat surface, and the device renders characters and objects as if they were on the surface. *Id.*

37. Multiple unique patterns can be placed around an environment; so long as one is always in view, the camera can maintain an estimate of the orientation and position. Gans Decl. ¶ 24. In this way, it can be used for navigation. *Id.* The necessity of placing patterns would make this approach useless for a majority of applications, particularly outdoors. *Id.* The camera would also need to remain on at all times, which would cause severe battery drain. *Id.*

38. Orientation of the camera can also be estimated over an indefinite amount of time using vision algorithms known as visual odometry. Gans Decl. ¶ 25. In visual odometry, changes in the image over time are used to estimate the camera velocity. *Id.* This velocity can be integrated over time to estimate the change in orientation. While these methods are well understood, they can only track change in relative orientation, not give absolute orientation. *Id.* They also require the camera to be on at all times, which will greatly reduce battery life. *Id.*

The Prior Art.

39. As noted in both the ’438 and ’978 patents, prior art portable electronic devices, such as pointing devices, smartphones and navigation equipment, had several deficiencies in addressing the technological challenges of mapping and transforming movement in a 3D space to a 2D display. These prior art portable devices could only output the movement of the device in 2D, rather than the 3D reference frame of the ’438 and’978 patents. ’438 patent 2:47-55; ’978 patent 2:41-58. In addition, the portable devices could not accurately calculate and account for movements of the device in a dynamic environment, such as erroneous drift measurements of the device or accelerations along with the direction of gravity. ’438 patent 2:55-62; ’978 patent 2:58-66. These prior art portable devices were also limited to detecting gravitational acceleration detected by the accelerometer, and were therefore incapable of accurately outputting the actual yaw, pitch and roll angles. ’438 patent 2:62-3:5; ’978 patent 2:66-3:13. Finally, for the specific case of pointing devices, when they extended beyond the border or boundary of the display, the absolute movement pattern was not mapped, but instead the location outside the boundary was ignored and a relative movement pattern used, which resulted in uncompensated errors. ’438 patent 3:16-51; ’978 patent 3:20-52.

PATENT INFRINGEMENT OF U.S. PATENT NO. 8,441,438

40. Plaintiff repeats and re-alleges each and every allegation of paragraphs 1-39 as though fully set forth herein.

41. The ’438 patent, titled “3D Pointing Device and Method for Compensating Movement Thereof,” was duly and legally issued by the United States Patent and Trademark Office on May 14, 2013 to CyWee Group Limited, as assignee of named inventors Zhou Ye, Chin-Lung Li, and Shun-Nan Liou.

42. CyWee is the owner of all right, title, and interest in and to the ’438 patent with full right to bring suit to enforce the patent, including the right to recover for past infringement damages.

43. CyWee is the owner of all right, title, and interest in and to the ’438 patent with full right to bring suit to enforce the patent, including the right to recover for past infringement damages.

44. The ’438 patent claims, *inter alia*, a machine capable of detecting, measuring, and calculating the movements and rotations of the machine—utilizing, *inter alia*, a six-axis motion sensor module, a data transmitting unit, and a computing processor in one or more claimed configurations—and methods for measuring and calculating the movements and rotations of a device within a spatial reference frame.

45. The ’438 patent is directed to useful and novel particular embodiments and methods for detecting, measuring, and calculating motion within a spatial reference frame. *See* Gans. Decl., ¶ 27. Specifically, the ’438 patent claims a novel system involving multiple sensor types and a novel method for using those sensors to overcome the limitations of the individual sensor types in accurately determining the orientation of a device. *See id.* ¶¶ 26-28. The ’438 patent is not intended to, and does not, claim every possible means of detecting, measuring, and calculating motion within a spatial reference frame. There are alternative methods to determining orientation within a spatial reference frame, such as systems and methods utilizing computer vision algorithms and/or cameras. *See id.* ¶¶ 22-25, 33. The ’438 patent is directed to a technological solution to a technological problem. *Id.* ¶¶ 33-35. Accordingly, the ’438 patent is not directed to, and does not claim, the mere concept of motion sensing or of detecting, measuring, and calculating motion within a spatial reference frame. *Id.* ¶¶ 29-35.

46. Each and every claim of the ’438 patent is valid and enforceable and each enjoys a statutory presumption of validity separate, apart, and in addition to the

statutory presumption of validity enjoyed by every other of its claims. 35 U.S.C. § 282.

47. CyWee is informed and believes, and thereupon alleges, that LG has been, and is currently directly or indirectly infringing one or more claims of the ’438 patent in violation of 35 U.S.C. § 271, including as stated below.

48. CyWee is informed and believes, and thereupon alleges, that LG has directly infringed, literally and/or under the doctrine of equivalents, and will continue to directly infringe claims of the ’438 patent by making, using, selling, offering to sell, and/or importing into the United States products that embody or practice the apparatus and/or method covered by one or more claims of the ’438 patent, including but not limited to Defendants’ devices such as LG V20, LG Stylo 3, LG Stylo 3 Plus, LG G5, LG G6, LG X Mach, LG X Venture, LG X Power 2, and LG X Cam (collectively referred to as “’438 Accused Products”).

49. CyWee adopts, and incorporates by reference, as if fully stated herein, the attached claim chart for claim 14 of the ’438 patent, which is attached hereto as Exhibit A. The claim chart describes and demonstrates how LG infringes the ’438 patent. In addition, CyWee alleges that LG infringes one or more additional claims of the ’438 patent in a similar manner.

50. Defendants’ acts of infringement have caused and will continue to cause substantial and irreparable damage to CyWee.

51. As a result of Defendants’ infringement of the ’438 patent, CyWee has been damaged. CyWee is, therefore, entitled to such damages pursuant to 35 U.S.C. § 284 in an amount that presently cannot be pled but that will be determined at trial.

52. The LG V20 includes a display screen.

53. The LG V20 includes a housing.

54. The LG V20 includes a 3-axis accelerometer.

55. The LG V20 includes a 3-axis gyroscope.

56. The LG V20 includes at least one printed circuit board ("PCB").

57. The LG V20 includes a 3-axis accelerometer attached to a PCB.

58. The LG V20 includes a 3-axis gyroscope attached to a PCB.

59. The LG V20 includes a 3-axis accelerometer that is capable of measuring accelerations.

60. The LG V20 includes a 3-axis gyroscope that is capable of measuring rotation rates.

61. The LG V20 includes a 3-axis accelerometer that is capable of measuring accelerations using a "Sensor Coordinate System" as described in the Android developer library. *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

62. The LG V20 includes a 3-axis gyroscope that is capable of measuring rotation rates using a "Sensor Coordinate System."

63. The LG V20 includes a processor that is capable of processing data associated with measurement from a 3-axis accelerometer.

64. The LG V20 includes a processor that is capable of processing data associated with measurement from a 3-axis gyroscope.

65. The LG V20 runs an Android operating system.

66. The Android operating system that runs on the LG V20 uses the measurement from a 3-axis accelerometer included in the device.

67. The Android operating system that runs on the LG V20 uses the measurement from a 3-axis gyroscope included in the device.

68. The Android operating system that runs on the LG V20 uses the measurement from a 3-axis accelerometer and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

69. The LG Stylo 3 includes a display screen.

70. The LG Stylo 3 includes a housing.

71. The LG Stylo 3 includes a 3-axis accelerometer.

72. The LG Stylo 3 includes a 3-axis gyroscope.

73. The LG Stylo 3 includes at least one printed circuit board ("PCB").

74. The LG Stylo 3 includes a 3-axis accelerometer attached to a PCB.

75. The LG Stylo 3 includes a 3-axis gyroscope attached to a PCB.

76. The LG Stylo 3 includes a 3-axis accelerometer that is capable of measuring accelerations.

77. The LG Stylo 3 includes a 3-axis gyroscope that is capable of measuring rotation rates.

78. The LG Stylo 3 includes a 3-axis accelerometer that is capable of measuring accelerations using a "Sensor Coordinate System" as described in the Android developer library. *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

79. The LG Stylo 3 includes a 3-axis gyroscope that is capable of measuring rotation rates using a "Sensor Coordinate System."

80. The LG Stylo 3 includes a processor that is capable of processing data associated with measurement from a 3-axis accelerometer.

81. The LG Stylo 3 includes a processor that is capable of processing data associated with measurement from a 3-axis gyroscope.

82. The LG Stylo 3 runs an Android operating system.

83. The Android operating system that runs on the LG Stylo 3 uses the measurement from a 3-axis accelerometer included in the device.

84. The Android operating system that runs on the LG Stylo 3 uses the measurement from a 3-axis gyroscope included in the device.

85. The Android operating system that runs on the LG Stylo 3 uses the

measurement from a 3-axis accelerometer and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

86. The LG Stylo 3 Plus includes a display screen.

87. The LG Stylo 3 Plus includes a housing.

88. The LG Stylo 3 Plus includes a 3-axis accelerometer.

89. The LG Stylo 3 Plus includes a 3-axis gyroscope.

90. The LG Stylo 3 Plus includes at least one printed circuit board ("PCB").

91. The LG Stylo 3 Plus includes a 3-axis accelerometer attached to a PCB.

92. The LG Stylo 3 Plus includes a 3-axis gyroscope attached to a PCB.

93. The LG Stylo 3 Plus includes a 3-axis accelerometer that is capable of measuring accelerations.

94. The LG Stylo 3 Plus includes a 3-axis gyroscope that is capable of measuring rotation rates.

95. The LG Stylo 3 Plus includes a 3-axis accelerometer that is capable of measuring accelerations using a "Sensor Coordinate System" as described in the Android developer library. *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

96. The LG Stylo 3 Plus includes a 3-axis gyroscope that is capable of measuring rotation rates using a "Sensor Coordinate System."

97. The LG Stylo 3 Plus includes a processor that is capable of processing data associated with measurement from a 3-axis accelerometer.

98. The LG Stylo 3 Plus includes a processor that is capable of processing data associated with measurement from a 3-axis gyroscope.

99. The LG Stylo 3 Plus runs an Android operating system.

100. The Android operating system that runs on the LG Stylo 3 Plus uses the measurement from a 3-axis accelerometer included in the device.

101. The Android operating system that runs on the LG Stylo 3 Plus uses the measurement from a 3-axis gyroscope included in the device.

102. The Android operating system that runs on the LG Stylo 3 Plus uses the measurement from a 3-axis accelerometer and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

103. The LG G5 includes a display screen.

104. The LG G5 includes a housing.

105. The LG G5 includes a 3-axis accelerometer.

106. The LG G5 includes a 3-axis gyroscope.

107. The LG G5 includes at least one printed circuit board ("PCB").

108. The LG G5 includes a 3-axis accelerometer attached to a PCB.

109. The LG G5 includes a 3-axis gyroscope attached to a PCB.

110. The LG G5 includes a 3-axis accelerometer that is capable of measuring accelerations.

111. The LG G5 includes a 3-axis gyroscope that is capable of measuring rotation rates.

112. The LG G5 includes a 3-axis accelerometer that is capable of measuring accelerations using a "Sensor Coordinate System" as described in the Android developer library. *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

113. The LG G5 includes a 3-axis gyroscope that is capable of measuring rotation rates using a "Sensor Coordinate System."

114. The LG G5 includes a processor that is capable of processing data associated with measurement from a 3-axis accelerometer.

115. The LG G5 includes a processor that is capable of processing data associated with measurement from a 3-axis gyroscope.

116. The LG G5 runs an Android operating system.

117. The Android operating system that runs on the LG G5 uses the measurement from a 3-axis accelerometer included in the device.

118. The Android operating system that runs on the LG G5 uses the measurement from a 3-axis gyroscope included in the device.

119. The Android operating system that runs on the LG G5 uses the measurement from a 3-axis accelerometer and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

120. The LG G6 includes a display screen.

121. The LG G6 includes a housing.

122. The LG G6 includes a 3-axis accelerometer.

123. The LG G6 includes a 3-axis gyroscope.

124. The LG G6 includes at least one printed circuit board (“PCB”).

125. The LG G6 includes a 3-axis accelerometer attached to a PCB.

126. The LG G6 includes a 3-axis gyroscope attached to a PCB.

127. The LG G6 includes a 3-axis accelerometer that is capable of measuring accelerations.

128. The LG G6 includes a 3-axis gyroscope that is capable of measuring rotation rates.

129. The LG G6 includes a 3-axis accelerometer that is capable of measuring accelerations using a “Sensor Coordinate System” as described in the Android developer library. *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing “Sensor Coordinate System”).

130. The LG G6 includes a 3-axis gyroscope that is capable of measuring rotation rates using a “Sensor Coordinate System.”

131. The LG G6 includes a processor that is capable of processing data

associated with measurement from a 3-axis accelerometer.

132. The LG G6 includes a processor that is capable of processing data associated with measurement from a 3-axis gyroscope.

133. The LG G6 runs an Android operating system.

134. The Android operating system that runs on the LG G6 uses the measurement from a 3-axis accelerometer included in the device.

135. The Android operating system that runs on the LG G6 uses the measurement from a 3-axis gyroscope included in the device.

136. The Android operating system that runs on the LG G6 uses the measurement from a 3-axis accelerometer and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

137. The LG X Mach includes a display screen.

138. The LG X Mach includes a housing.

139. The LG X Mach includes a 3-axis accelerometer.

140. The LG X Mach includes a 3-axis gyroscope.

141. The LG X Mach includes at least one printed circuit board (“PCB”).

142. The LG X Mach includes a 3-axis accelerometer attached to a PCB.

143. The LG X Mach includes a 3-axis gyroscope attached to a PCB.

144. The LG X Mach includes a 3-axis accelerometer that is capable of measuring accelerations.

145. The LG X Mach includes a 3-axis gyroscope that is capable of measuring rotation rates.

146. The LG X Mach includes a 3-axis accelerometer that is capable of measuring accelerations using a “Sensor Coordinate System” as described in the Android developer library. *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing “Sensor Coordinate System”).

147. The LG X Mach includes a 3-axis gyroscope that is capable of measuring rotation rates using a "Sensor Coordinate System."

148. The LG X Mach includes a processor that is capable of processing data associated with measurement from a 3-axis accelerometer.

149. The LG X Mach includes a processor that is capable of processing data associated with measurement from a 3-axis gyroscope.

150. The LG X Mach runs an Android operating system.

151. The Android operating system that runs on the LG X Mach uses the measurement from a 3-axis accelerometer included in the device.

152. The Android operating system that runs on the LG X Mach uses the measurement from a 3-axis gyroscope included in the device.

153. The Android operating system that runs on the LG X Mach uses the measurement from a 3-axis accelerometer and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

154. The LG X Venture includes a display screen.

155. The LG X Venture includes a housing.

156. The LG X Venture includes a 3-axis accelerometer.

157. The LG X Venture includes a 3-axis gyroscope.

158. The LG X Venture includes at least one printed circuit board ("PCB").

159. The LG X Venture includes a 3-axis accelerometer attached to a PCB.

160. The LG X Venture includes a 3-axis gyroscope attached to a PCB.

161. The LG X Venture includes a 3-axis accelerometer that is capable of measuring accelerations.

162. The LG X Venture includes a 3-axis gyroscope that is capable of measuring rotation rates.

163. The LG X Venture includes a 3-axis accelerometer that is capable of measuring accelerations using a "Sensor Coordinate System" as described in the

Android developer library. *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

164. The LG X Venture includes a 3-axis gyroscope that is capable of measuring rotation rates using a "Sensor Coordinate System."

165. The LG X Venture includes a processor that is capable of processing data associated with measurement from a 3-axis accelerometer.

166. The LG X Venture includes a processor that is capable of processing data associated with measurement from a 3-axis gyroscope.

167. The LG X Venture runs an Android operating system.

168. The Android operating system that runs on the LG X Venture uses the measurement from a 3-axis accelerometer included in the device.

169. The Android operating system that runs on the LG X Venture uses the measurement from a 3-axis gyroscope included in the device.

170. The Android operating system that runs on the LG X Venture uses the measurement from a 3-axis accelerometer and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

171. The LG X Power 2 includes a display screen.

172. The LG X Power 2 includes a housing.

173. The LG X Power 2 includes a 3-axis accelerometer.

174. The LG X Power 2 includes a 3-axis gyroscope.

175. The LG X Power 2 includes at least one printed circuit board ("PCB").

176. The LG X Power 2 includes a 3-axis accelerometer attached to a PCB.

177. The LG X Power 2 includes a 3-axis gyroscope attached to a PCB.

178. The LG X Power 2 includes a 3-axis accelerometer that is capable of measuring accelerations.

179. The LG X Power 2 includes a 3-axis gyroscope that is capable of

measuring rotation rates.

180. The LG X Power 2 includes a 3-axis accelerometer that is capable of measuring accelerations using a “Sensor Coordinate System” as described in the Android developer library. *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing “Sensor Coordinate System”).

181. The LG X Power 2 includes a 3-axis gyroscope that is capable of measuring rotation rates using a “Sensor Coordinate System.”

182. The LG X Power 2 includes a processor that is capable of processing data associated with measurement from a 3-axis accelerometer.

183. The LG X Power 2 includes a processor that is capable of processing data associated with measurement from a 3-axis gyroscope.

184. The LG X Power 2 runs an Android operating system.

185. The Android operating system that runs on the LG X Power 2 uses the measurement from a 3-axis accelerometer included in the device.

186. The Android operating system that runs on the LG X Power 2 uses the measurement from a 3-axis gyroscope included in the device.

187. The Android operating system that runs on the LG X Power 2 uses the measurement from a 3-axis accelerometer and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

188. The LG X Cam includes a display screen.

189. The LG X Cam includes a housing.

190. The LG X Cam includes a 3-axis accelerometer.

191. The LG X Cam includes a 3-axis gyroscope.

192. The LG X Cam includes at least one printed circuit board (“PCB”).

193. The LG X Cam includes a 3-axis accelerometer attached to a PCB.

194. The LG X Cam includes a 3-axis gyroscope attached to a PCB.

195. The LG X Cam includes a 3-axis accelerometer that is capable of measuring accelerations.

196. The LG X Cam includes a 3-axis gyroscope that is capable of measuring rotation rates.

197. The LG X Cam includes a 3-axis accelerometer that is capable of measuring accelerations using a “Sensor Coordinate System” as described in the Android developer library. *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing “Sensor Coordinate System”).

198. The LG X Cam includes a 3-axis gyroscope that is capable of measuring rotation rates using a “Sensor Coordinate System.”

199. The LG X Cam includes a processor that is capable of processing data associated with measurement from a 3-axis accelerometer.

200. The LG X Cam includes a processor that is capable of processing data associated with measurement from a 3-axis gyroscope.

201. The LG X Cam runs an Android operating system.

202. The Android operating system that runs on the LG X Cam uses the measurement from a 3-axis accelerometer included in the device.

203. The Android operating system that runs on the LG X Cam uses the measurement from a 3-axis gyroscope included in the device.

204. The Android operating system that runs on the LG X Cam uses the measurement from a 3-axis accelerometer and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

PATENT INFRINGEMENT OF U.S. PATENT NO. 8,552,978

205. Plaintiff repeats and re-alleges each and every allegation of paragraphs 1-204 as though fully set forth herein.

206. The ’978 patent, titled “3D Pointing Device and Method for Compensating Rotations of the 3D Pointing Device Thereof,” was duly and legally issued by the United States Patent and Trademark Office on October 8, 2013 to CyWee Group Limited, as assignee of named inventors Zhou Ye, Chin-Lung Li, and Shun-Nan Liou.

207. CyWee is the owner of all right, title, and interest in and to the ’978 patent with full right to bring suit to enforce the patent, including the right to recover for past infringement damages.

208. The ’978 patent claims, *inter alia*, a machine capable of detecting, measuring, and calculating the movements and rotations of the machine—utilizing, *inter alia*, a nine-axes motion sensor module and two computing processors in one or more claimed configurations—and methods for measuring and calculating the movements and rotations of a device within a spatial reference frame. *See* Gans Decl. ¶¶ 8-12.

209. The ’978 patent is directed to useful and novel particular embodiments and methods for detecting, measuring, and calculating motion within a spatial reference frame. *Id.* ¶ 27. Specifically, the ’978 patent claims a novel system involving multiple sensor types and a novel method for using those sensors to overcome the limitations of the individual sensor types in accurately determining the orientation of a device. *See id.* ¶¶ 26-28. The ’978 patent is not intended to, and does not, claim every possible means of detecting, measuring, and calculating motion within a spatial reference frame. There are alternative methods to determining orientation within a spatial reference frame, such as systems and methods utilizing computer vision algorithms and/or cameras. *See id.* ¶¶ 22-25, 33. The ’978 patent is directed to a technological solution to a technological problem. *Id.* ¶¶ 33-35. Accordingly, the ’978 patent is not directed to, and does not claim, the mere concept of motion sensing or of detecting, measuring, and calculating motion within a spatial reference frame. *Id.* ¶¶

29-35.

210. Each and every claim of the ’978 patent is valid and enforceable and each enjoys a statutory presumption of validity separate, apart, and in addition to the statutory presumption of validity enjoyed by every other of its claims. 35 U.S.C. § 282.

211. CyWee is informed and believes, and thereupon alleges, that LG has been, and is currently directly and/or indirectly infringing one or more claims of the ’978 patent in violation of 35 U.S.C. § 271, including as stated below.

212. CyWee is informed and believes, and thereupon alleges, that LG has directly infringed, literally and/or under the doctrine of equivalents, and will continue to directly infringe claims of the ’978 patent by making, using, selling, offering to sell, and/or importing into the United States products that embody or practice the apparatus and/or method covered by one or more claims of the ’978 patent, including but not limited to Defendants’ devices such as LG V20, LG Stylo 3, LG Stylo 3 Plus, LG G5, LG G6, LG X mach, LG X venture, LG X power 2, and LG X cam (collectively referred to as “’978 Accused Products”).

213. CyWee adopts, and incorporates by reference, as if fully stated herein, the attached claim chart for claim 10 of the ’978 patent, which is attached hereto as Exhibit B. The claim chart describes and demonstrates how LG infringes the ’978 patent. In addition, CyWee alleges that LG infringes one or more additional claims of the ’978 patent in a similar manner.

214. Defendants’ acts of infringement have caused and will continue to cause substantial and irreparable damage to CyWee.

215. As a result of Defendants’ infringement of the ’978 patent, CyWee has been damaged. CyWee is, therefore, entitled to such damages pursuant to 35 U.S.C. § 284 in an amount that presently cannot be pled but that will be determined at trial.

216. The LG V20 includes a 3-axis geomagnetic sensor.

217. The LG V20 includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

218. The LG V20 includes a 3-axis geomagnetic field sensor to measure a geomagnetic field using a "Sensor Coordinate System." *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

219. The Android operating system that runs on the LG V20 uses the measurement from a 3-axis geomagnetic sensor included in the device.

220. The Android operating system that runs on the LG V20 uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

221. The Android operating system that runs on the LG V20 uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device that can be represented by an azimuth angle, a pitch angle, and a roll angle.

222. The LG V20 has the ability to directly control apps by moving or rotating the device (for example, racing game apps).

223. The LG V20 has the ability to run apps that can provide information based on the direction your device is facing, such as a map or navigation app.

224. The LG V20 includes a 3-axis geomagnetic sensor.

225. The LG V20 includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

226. The LG Stylo 3 includes a 3-axis geomagnetic sensor.

227. The LG Stylo 3 includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

228. The LG Stylo 3 includes a 3-axis geomagnetic field sensor to measure a geomagnetic field using a "Sensor Coordinate System." *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

229. The Android operating system that runs on the LG Stylo 3 uses the measurement from a 3-axis geomagnetic sensor included in the device.

230. The Android operating system that runs on the LG Stylo 3 uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

231. The Android operating system that runs on the LG Stylo 3 uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device that can be represented by an azimuth angle, a pitch angle, and a roll angle.

232. The LG Stylo 3 has the ability to directly control apps by moving or rotating the device (for example, racing game apps).

233. The LG Stylo 3 has the ability to run apps that can provide information based on the direction your device is facing, such as a map or navigation app.

234. The LG Stylo 3 includes a 3-axis geomagnetic sensor.

235. The LG Stylo 3 includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

236. The LG Stylo 3 Plus includes a 3-axis geomagnetic sensor.

237. The LG Stylo 3 Plus includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

238. The LG Stylo 3 Plus includes a 3-axis geomagnetic field sensor to measure a geomagnetic field using a "Sensor Coordinate System." *See*

https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

239. The Android operating system that runs on the LG Stylo 3 Plus uses the measurement from a 3-axis geomagnetic sensor included in the device.

240. The Android operating system that runs on the LG Stylo 3 Plus uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

241. The Android operating system that runs on the LG Stylo 3 Plus uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device that can be represented by an azimuth angle, a pitch angle, and a roll angle.

242. The LG Stylo 3 Plus has the ability to directly control apps by moving or rotating the device (for example, racing game apps).

243. The LG Stylo 3 Plus has the ability to run apps that can provide information based on the direction your device is facing, such as a map or navigation app.

244. The LG Stylo 3 Plus includes a 3-axis geomagnetic sensor.

245. The LG Stylo 3 Plus includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

246. The LG G5 includes a 3-axis geomagnetic sensor.

247. The LG G5 includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

248. The LG G5 includes a 3-axis geomagnetic field sensor to measure a geomagnetic field using a "Sensor Coordinate System." *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-

coords (describing “Sensor Coordinate System”).

249. The Android operating system that runs on the LG G5 uses the measurement from a 3-axis geomagnetic sensor included in the device.

250. The Android operating system that runs on the LG G5 uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

251. The Android operating system that runs on the LG G5 uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device that can be represented by an azimuth angle, a pitch angle, and a roll angle.

252. The LG G5 has the ability to directly control apps by moving or rotating the device (for example, racing game apps).

253. The LG G5 has the ability to run apps that can provide information based on the direction your device is facing, such as a map or navigation app.

254. The LG G5 includes a 3-axis geomagnetic sensor.

255. The LG G5 includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

256. The LG G6 includes a 3-axis geomagnetic sensor.

257. The LG G6 includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

258. The LG G6 includes a 3-axis geomagnetic field sensor to measure a geomagnetic field using a “Sensor Coordinate System.” *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing “Sensor Coordinate System”).

259. The Android operating system that runs on the LG G6 uses the

measurement from a 3-axis geomagnetic sensor included in the device.

260. The Android operating system that runs on the LG G6 uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

261. The Android operating system that runs on the LG G6 uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device that can be represented by an azimuth angle, a pitch angle, and a roll angle.

262. The LG G6 has the ability to directly control apps by moving or rotating the device (for example, racing game apps).

263. The LG G6 has the ability to run apps that can provide information based on the direction your device is facing, such as a map or navigation app.

264. The LG G6 includes a 3-axis geomagnetic sensor.

265. The LG G6 includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

266. The LG X Mach includes a 3-axis geomagnetic sensor.

267. The LG X Mach includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

268. The LG X Mach includes a 3-axis geomagnetic field sensor to measure a geomagnetic field using a "Sensor Coordinate System." *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

269. The Android operating system that runs on the LG X Mach uses the measurement from a 3-axis geomagnetic sensor included in the device.

270. The Android operating system that runs on the LG X Mach uses the

measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

271. The Android operating system that runs on the LG X Mach uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device that can be represented by an azimuth angle, a pitch angle, and a roll angle.

272. The LG X Mach has the ability to directly control apps by moving or rotating the device (for example, racing game apps).

273. The LG X Mach has the ability to run apps that can provide information based on the direction your device is facing, such as a map or navigation app.

274. The LG X Mach includes a 3-axis geomagnetic sensor.

275. The LG X Mach includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

276. The LG X Venture includes a 3-axis geomagnetic sensor.

277. The LG X Venture includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

278. The LG X Venture includes a 3-axis geomagnetic field sensor to measure a geomagnetic field using a "Sensor Coordinate System." *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

279. The Android operating system that runs on the LG X Venture uses the measurement from a 3-axis geomagnetic sensor included in the device.

280. The Android operating system that runs on the LG X Venture uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate

an attitude of the device.

281. The Android operating system that runs on the LG X Venture uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device that can be represented by an azimuth angle, a pitch angle, and a roll angle.

282. The LG X Venture has the ability to directly control apps by moving or rotating the device (for example, racing game apps).

283. The LG X Venture has the ability to run apps that can provide information based on the direction your device is facing, such as a map or navigation app.

284. The LG X Venture includes a 3-axis geomagnetic sensor.

285. The LG X Venture includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

286. The LG X Power 2 includes a 3-axis geomagnetic sensor.

287. The LG X Power 2 includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

288. The LG X Power 2 includes a 3-axis geomagnetic field sensor to measure a geomagnetic field using a "Sensor Coordinate System." *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing "Sensor Coordinate System").

289. The Android operating system that runs on the LG X Power 2 uses the measurement from a 3-axis geomagnetic sensor included in the device.

290. The Android operating system that runs on the LG X Power 2 uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

291. The Android operating system that runs on the LG X Power 2 uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device that can be represented by an azimuth angle, a pitch angle, and a roll angle.

292. The LG X Power 2 has the ability to directly control apps by moving or rotating the device (for example, racing game apps).

293. The LG X Power 2 has the ability to run apps that can provide information based on the direction your device is facing, such as a map or navigation app.

294. The LG X Power 2 includes a 3-axis geomagnetic sensor.

295. The LG X Power 2 includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

296. The LG X Cam includes a 3-axis geomagnetic sensor.

297. The LG X Cam includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

298. The LG X Cam includes a 3-axis geomagnetic field sensor to measure a geomagnetic field using a “Sensor Coordinate System.” *See* https://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords (describing “Sensor Coordinate System”).

299. The Android operating system that runs on the LG X Cam uses the measurement from a 3-axis geomagnetic sensor included in the device.

300. The Android operating system that runs on the LG X Cam uses the measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device.

301. The Android operating system that runs on the LG X Cam uses the

measurement from a 3-axis accelerometer, the measurement from a 3-axis geomagnetic field sensor, and the measurement from a 3-axis gyroscope to calculate an attitude of the device that can be represented by an azimuth angle, a pitch angle, and a roll angle.

302. The LG X Cam has the ability to directly control apps by moving or rotating the device (for example, racing game apps).

303. The LG X Cam has the ability to run apps that can provide information based on the direction your device is facing, such as a map or navigation app.

304. The LG X Cam includes a 3-axis geomagnetic sensor.

305. The LG X Cam includes a 3-axis geomagnetic sensor that is capable of measuring a geomagnetic field.

PRAYER FOR RELIEF

WHEREFORE, Plaintiff prays for entry of judgment against Defendants as follows:

A. A judgment that Defendants have infringed and continue to infringe the ’438 patent and ’978 patent, directly and/or indirectly by way of inducing infringement of such patents as alleged herein;

B. That Defendants provide to CyWee an accounting of all gains, profits and advantages derived by Defendants’ infringement of the ’978 patent and ’438 patent, and that CyWee be awarded damages adequate to compensate them for the wrongful infringement by Defendants, including treble damages for willful infringement, in accordance with 35 U.S.C. § 284;

C. That CyWee be awarded any other supplemental damages and interest on all damages, including, but not limited to, attorney fees available under 35 U.S.C. § 285;

D. That the Court permanently enjoin Defendants and all those in privity with Defendants from making, having made, selling, offering for sale, distributing and/or using products that infringe the ’978 patent and/or the ’438 patent, including the ’978 Accused Products and/or ’438 Accused Products, in the United States; and

E. That CyWee be awarded such other and further relief and all remedies available at law.

DEMAND FOR JURY TRIAL

Pursuant to Federal Rule of Civil Procedure 38(b), CyWee hereby demands a trial by jury on all issues triable to a jury.

Dated: October 6, 2017

Respectfully submitted,

/s/ *Ari Rafilson*
Kent Walker
Lewis Kohn & Walker LLP
15030 Avenue of Science, Suite 201
San Diego CA 92128
Telephone: (858) 436-1330
Fax: (858) 436-1349

Michael W. Shore*
mshore@shorechan.com
Alfonso G. Chan*
achan@shorechan.com
Ari Rafilson*
arafilson@shorechan.com
Christopher L. Evans*
cevans@shorechan.com
Paul T. Beeler*
pbeeler@shorechan.com
SHORE CHAN DEPUMPO LLP
901 Main Street, Suite 3300
Dallas, California 75202
Telephone: (214) 593-9110
Facsimile: (214) 593-9111

* *Admitted Pro Hac Vice*

COUNSEL FOR PLAINTIFF CYWEE GROUP LTD.

CERTIFICATE OF SERVICE

I hereby certify that all counsel of record who have consented to electronic service are being served on October 6, 2017 with a copy of the foregoing document and exhibits thereto via the Court's CM/ECF system, per Local Rule CV-5.2 and Electronic Case Filing Administrative Policies and Procedures Manual, Section 2(d)(2).

s/ *Ari Rafilson*
Ari Rafilson

Attorney for Plaintiff
CYWEE GROUP LTD.

TABLE OF CONTENTS – EXHIBITS TO FIRST AMENDED COMPLAINT

TABLE OF CONTENTS – EXHIBITS TO FIRST AMENDED COMPLAINT

EXHIBIT A

U.S. Patent No. 8,441,438

LG V20

SUBJECT TO CHANGE

EXHIBIT A
PAGE 1

U.S. Patent No. 8,441,438 – LG V20

Claim 14

A method for obtaining a resulting deviation including resultant angles in a **spatial pointer reference frame** of a **three-dimensional (3D) pointing device** utilizing a six-axis motion sensor module therein and subject to movements and rotations in dynamic environments in said **spatial pointer reference frame**, comprising the steps of:

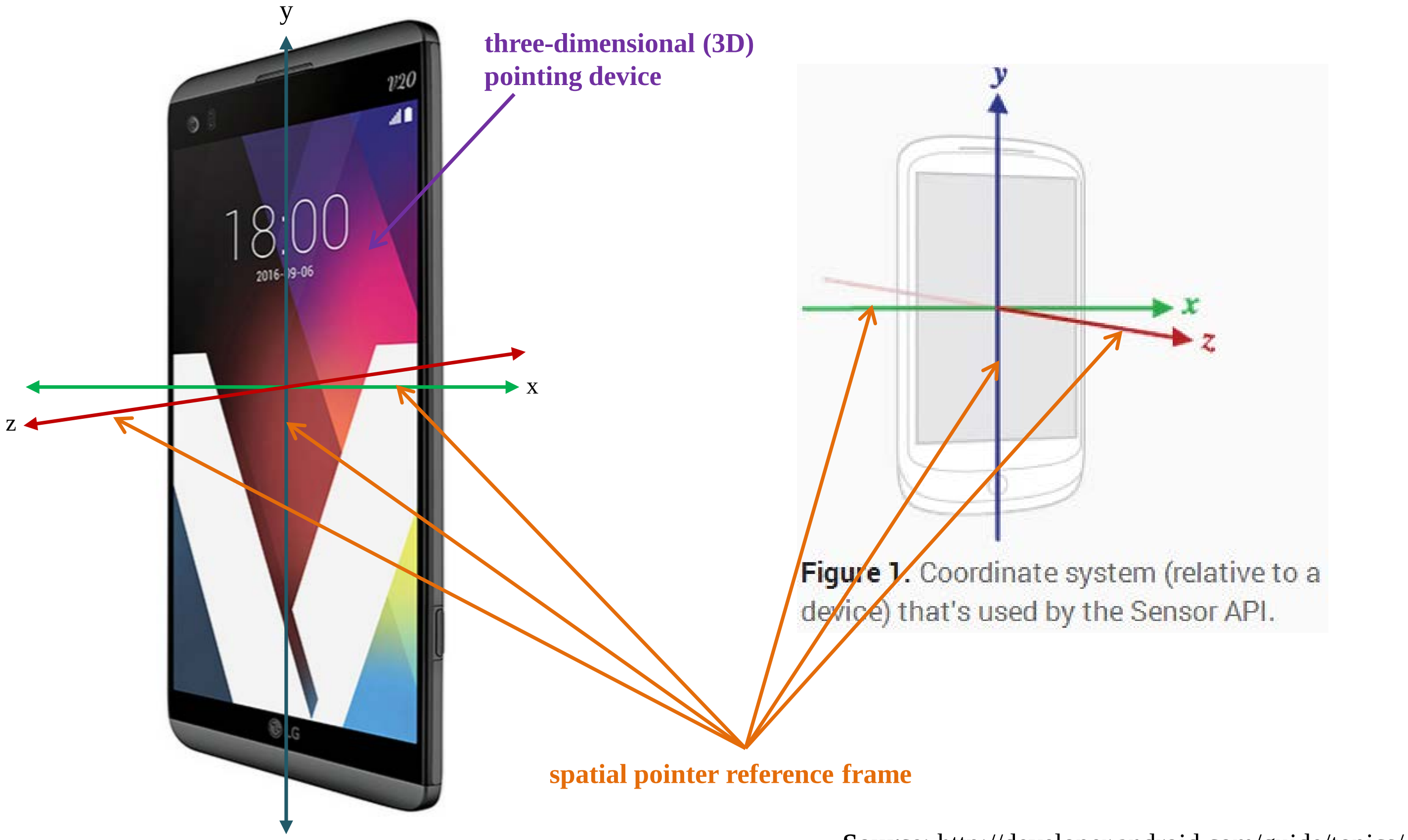
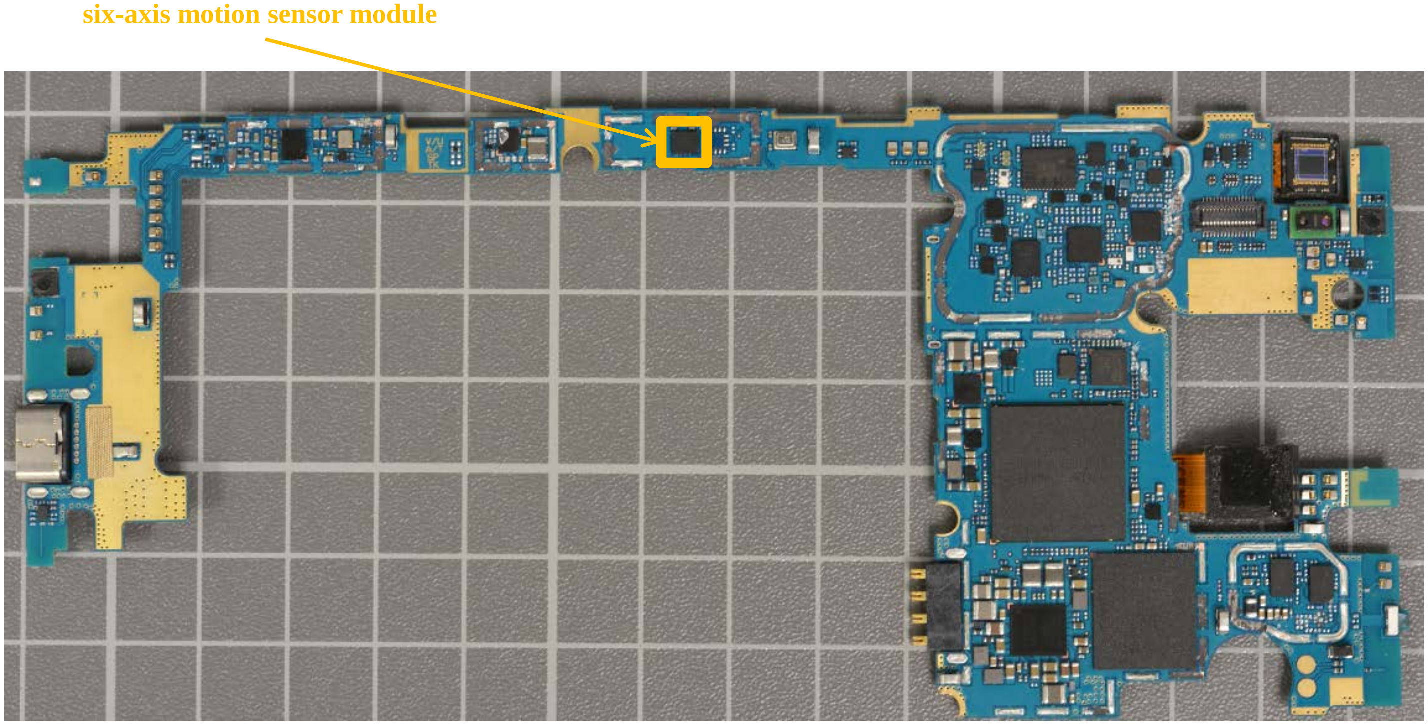


Figure 1. Coordinate system (relative to a device) that's used by the Sensor API.

Source: http://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords

U.S. Patent No. 8,441,438 – LG V20

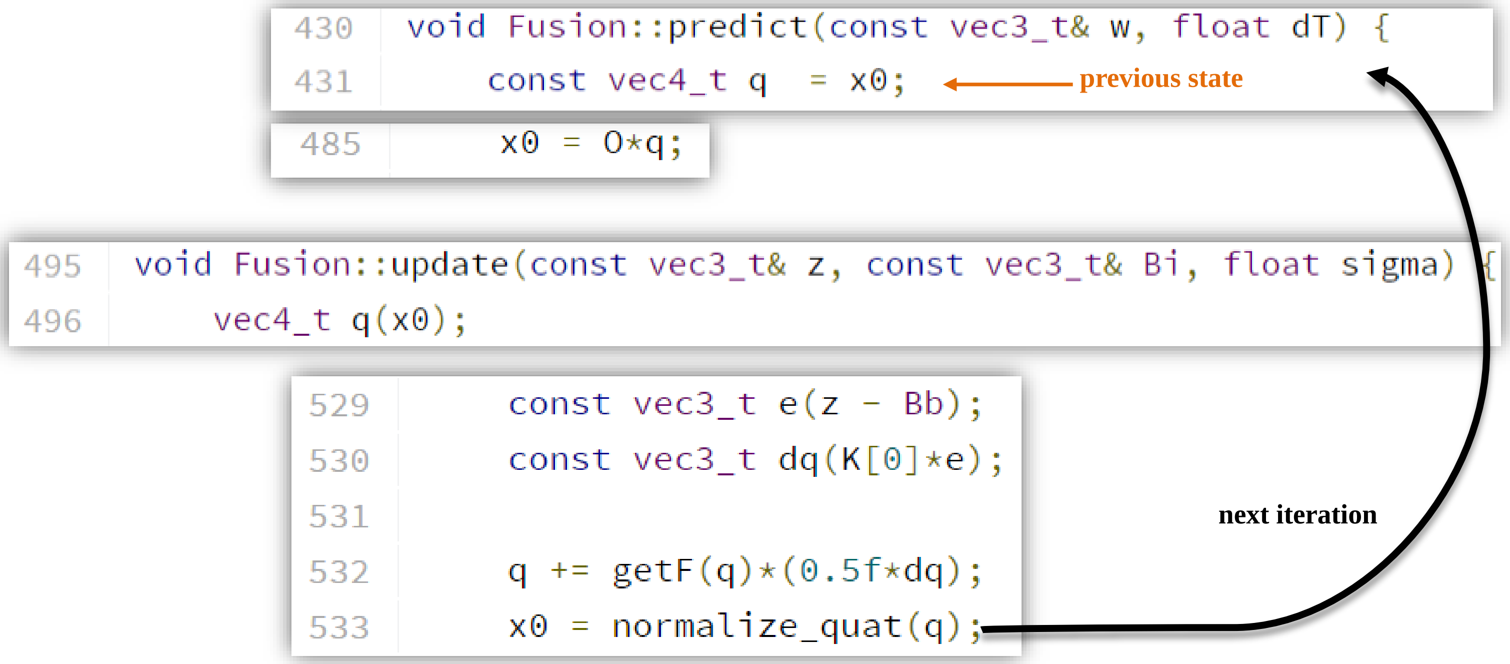
Claim 14
A method for obtaining a resulting deviation including resultant angles in a spatial pointer reference frame of a three-dimensional (3D) pointing device utilizing a **six-axis motion sensor module** therein and subject to movements and rotations in dynamic environments in said spatial pointer reference frame, comprising the steps of:



Claim 14

obtaining a **previous state** of the six-axis motion sensor module; wherein the **previous state** includes an initial-value set associated with **previous angular velocities** gained from the motion sensor signals of the six-axis motion sensor module at a previous time T−1;

The previous state is obtained through an update program that includes a `predict()` function and an `update()` function. Those functions that are used to update the global variable `x0` based on `x0` (the **previous state**) associated with **previous angular velocities** `w` gained at a previous time T-1 to obtain an updated state `x0`. The updated state `x0` becomes the previous state `x0` at time T (the next iteration) of the update program to obtain the updated state `x0` at time T.



```
void Fusion::predict(const vec3_t& w, float dT) {
    const vec4_t q  = x0;
```

```
    x0 = O*q;
```

```
void Fusion::update(const vec3_t& z, const vec3_t& Bi, float sigma) {
    vec4_t q(x0);
```

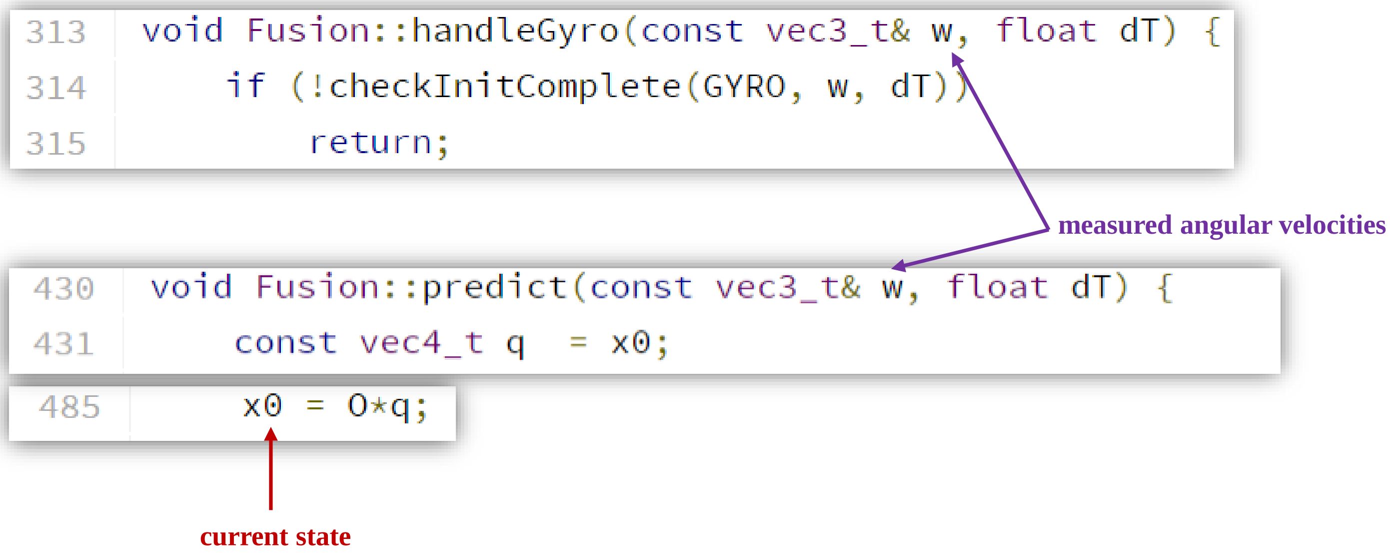
```
    const vec3_t e(z - Bb);
    const vec3_t dq(K[0]*e);

    q += getF(q)*(0.5f*dq);
    x0 = normalize_quat(q);
```

Source: https://android.googlesource.com/platform/frameworks/native/+/master/services/sensorservice/Fusion.cpp

U.S. Patent No. 8,441,438 – LG V20

Claim 14

obtaining a **current state** of the six-axis motion sensor module by obtaining **measured angular velocities** ω_x, ω_y, ω_z gained from the motion sensor signals of the six-axis motion sensor module at a current time T;



```
void Fusion::handleGyro(const vec3_t& w, float dT) {
    if (!checkInitComplete(GYRO, w, dT))
        return;
```

```
void Fusion::predict(const vec3_t& w, float dT) {
    const vec4_t q  = x0;
```

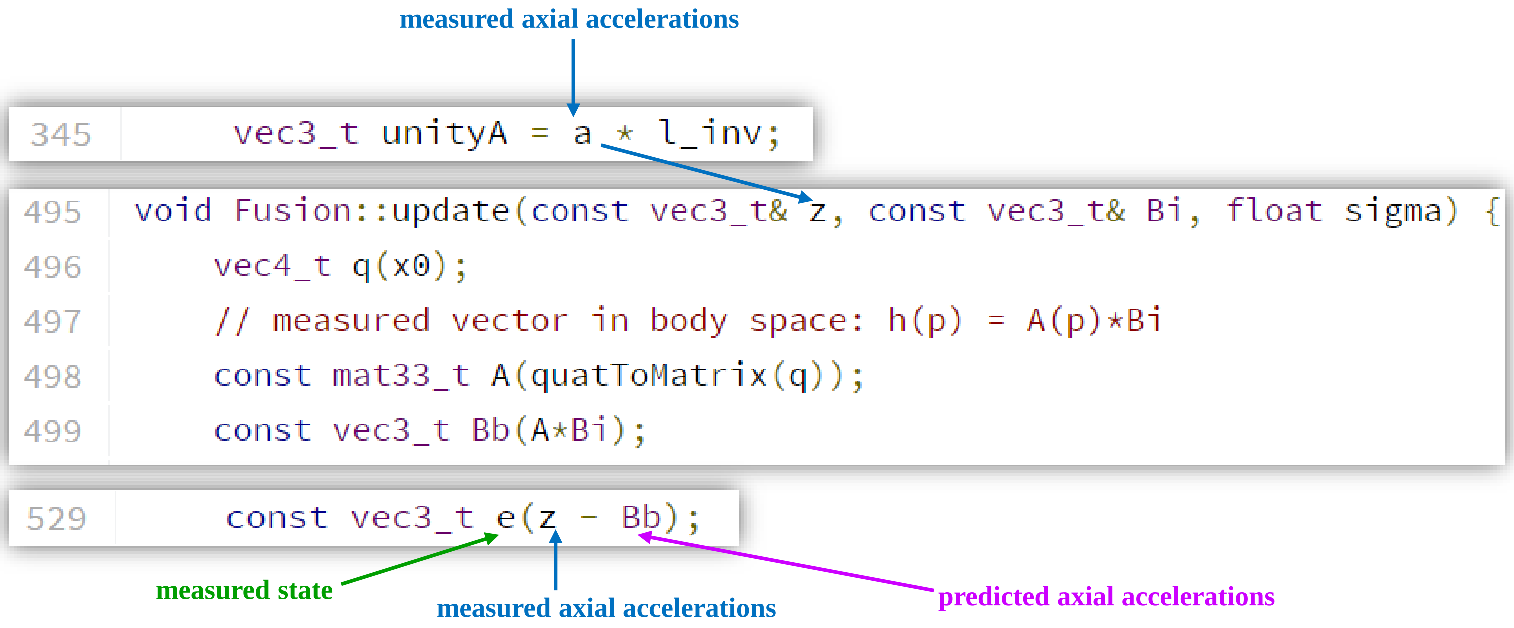
```
    x0 = O*q;
```

Source: https://android.googlesource.com/platform/frameworks/native/+/master/services/sensorservice/Fusion.cpp

U.S. Patent No. 8,441,438 – LG V20

Claim 14

obtaining a **measured state** of the six-axis motion sensor module by obtaining **measured axial accelerations** Ax, Ay, Az gained from the motion sensor signals of the six-axis motion sensor module at the current time T and calculating **predicted axial accelerations** Ax′, Ay′, Az′ based on the **measured angular velocities** ω_x, ω_y, ω_z of the current state of the six-axis motion sensor module **without using any derivatives of the measured angular velocities** ωx, ωy, ωz;



```
    vec3_t unityA = a * l_inv;
```

```
void Fusion::update(const vec3_t& z, const vec3_t& Bi, float sigma) {
    vec4_t q(x0);
    // measured vector in body space: h(p) = A(p)*Bi
    const mat33_t A(quatToMatrix(q));
    const vec3_t Bb(A*Bi);
```

```
    const vec3_t e(z - Bb);
```

As shown in the code above, the predicted measurement is obtained based on the first signal set **without using any derivatives of the measured angular velocities**.

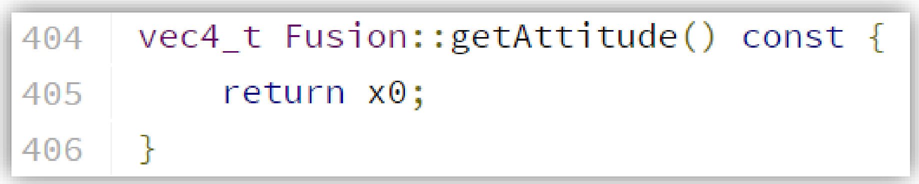
Source: https://android.googlesource.com/platform/frameworks/native/+/master/services/sensorservice/Fusion.cpp

U.S. Patent No. 8,441,438 – LG V20

Claim 14

said **current state** of the six-axis motion sensor module is a second quaternion with respect to said current time T;

As shown in the examples provided, the **current state** is represented by the global state variable `x0`, which is a quaternion with respect to the current time T.



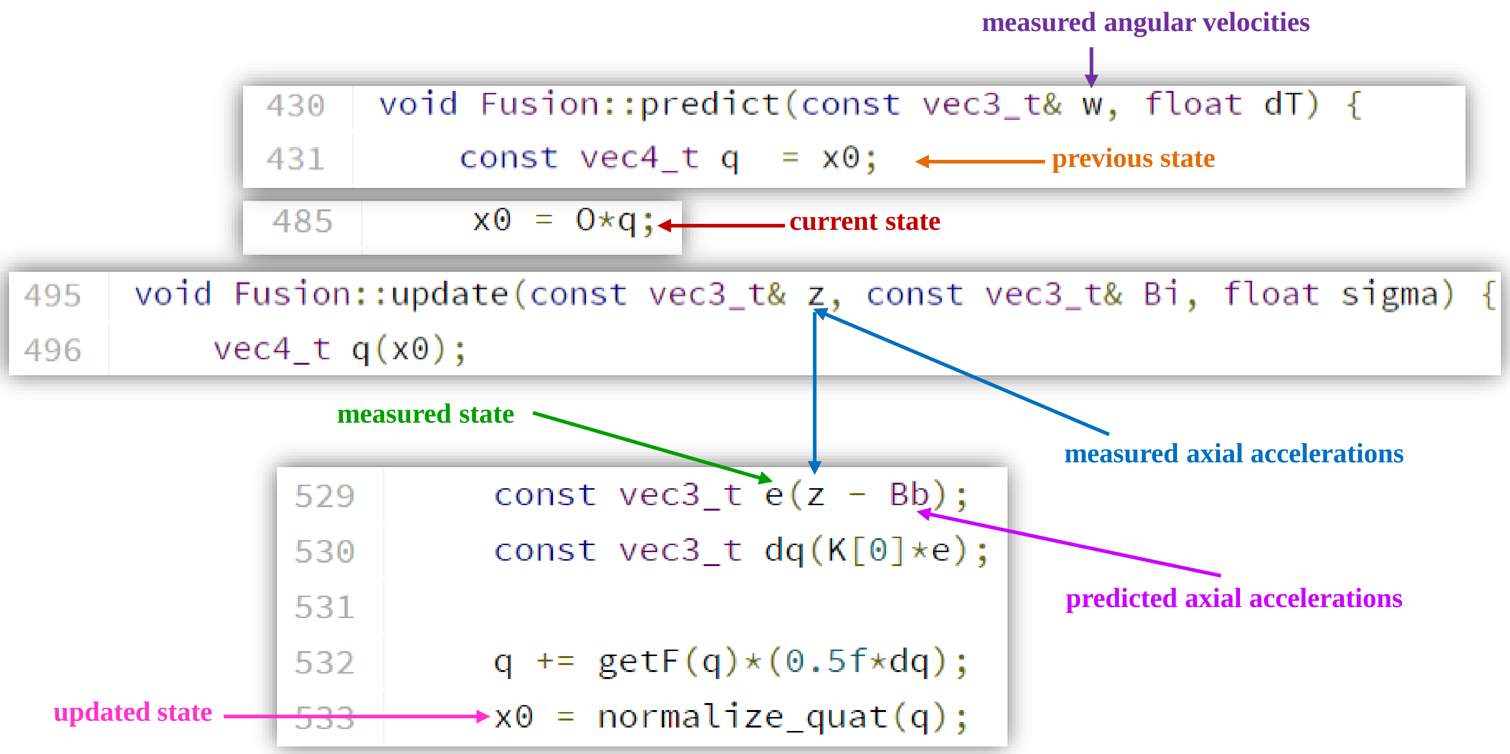
```
vec4_t Fusion::getAttitude() const {
    return x0;
}
```

Source: https://android.googlesource.com/platform/frameworks/native/+/master/services/sensorservice/Fusion.cpp

U.S. Patent No. 8,441,438 – LG V20

Claim 14

comparing the second quaternion in relation to the **measured angular velocities** ω_x, ω_y, ω_z of the **current state** at current time T with the **measured axial accelerations** Ax, Ay, Az and the **predicted axial accelerations** Ax′, Ay′, Az′ also at current time T; obtaining an **updated state** of the six-axis motion sensor module by comparing the **current state** with the **measured state** of the six-axis motion sensor module; and



```
void Fusion::predict(const vec3_t& w, float dT) {
    const vec4_t q  = x0;
```

```
    x0 = O*q;
```

```
void Fusion::update(const vec3_t& z, const vec3_t& Bi, float sigma) {
    vec4_t q(x0);
```

```
    const vec3_t e(z - Bb);
    const vec3_t dq(K[0]*e);

    q += getF(q)*(0.5f*dq);
    x0 = normalize_quat(q);
```

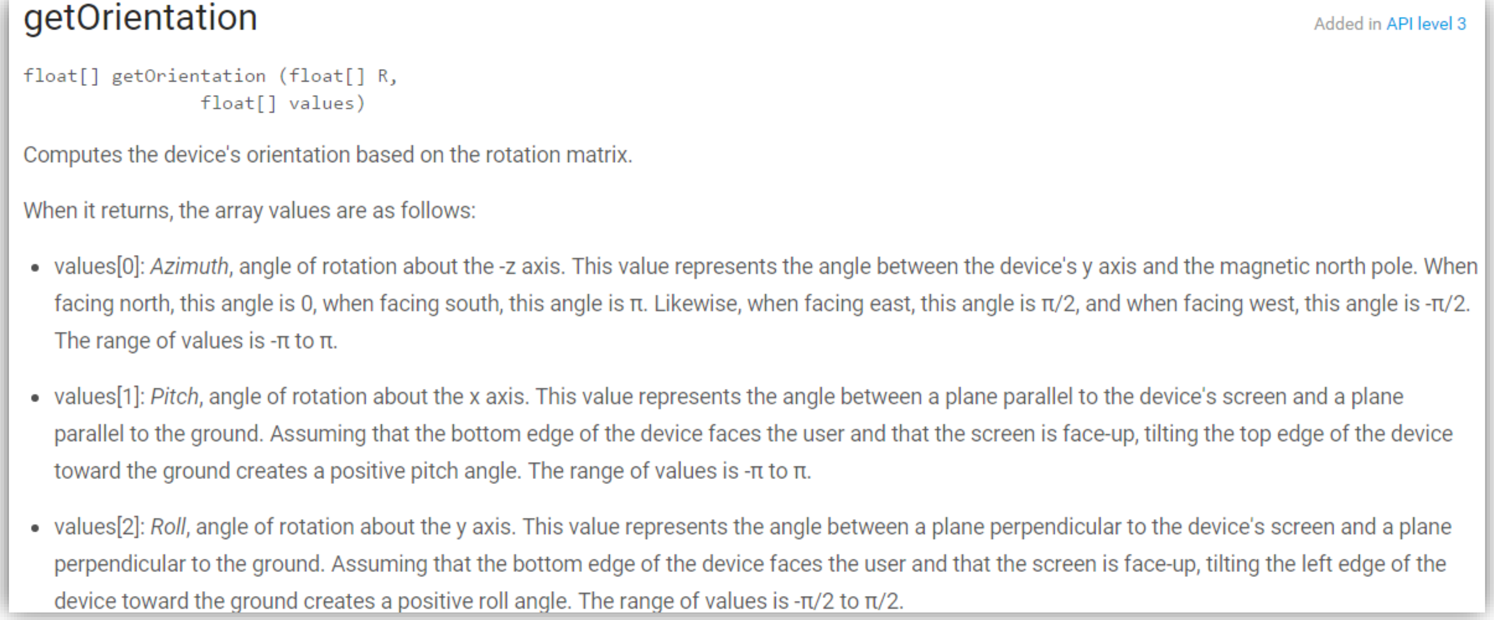
Source: https://android.googlesource.com/platform/frameworks/native/+/master/services/sensorservice/Fusion.cpp

Claim 14

calculating and converting the **updated state** of the six axis motion sensor module to said **resulting deviation comprising said resultant angles** in said spatial pointer reference frame of the 3D pointing device.

The **updated state** `x0` is in quaternion form, and can easily be converted to resultant angles.

According to Android's developer library, the `getOrientation()` function "computes the device's orientation based on the rotation matrix," and returns **resultant angles** including the Azimuth, Pitch, and Roll angles.



getOrientation

Added in API level 3

```
float[] getOrientation (float[] R,
                float[] values)
```

Computes the device's orientation based on the rotation matrix.

When it returns, the array values are as follows:

- values[0]: *Azimuth*, angle of rotation about the -z axis. This value represents the angle between the device's y axis and the magnetic north pole. When facing north, this angle is 0, when facing south, this angle is π. Likewise, when facing east, this angle is π/2, and when facing west, this angle is -π/2. The range of values is -π to π.
- values[1]: *Pitch*, angle of rotation about the x axis. This value represents the angle between a plane parallel to the device's screen and a plane parallel to the ground. Assuming that the bottom edge of the device faces the user and that the screen is face-up, tilting the top edge of the device toward the ground creates a positive pitch angle. The range of values is -π to π.
- values[2]: *Roll*, angle of rotation about the y axis. This value represents the angle between a plane perpendicular to the device's screen and a plane perpendicular to the ground. Assuming that the bottom edge of the device faces the user and that the screen is face-up, tilting the left edge of the device toward the ground creates a positive roll angle. The range of values is -π/2 to π/2.

The `getRotationMatrixFromVector()` function "convert[s] a rotation vector to a rotation matrix," and the `getQuaternionFromVector()` function "convert[s] a rotation vector to a normalized quaternion." Therefore, the quaternion, `x0`, can be easily converted to its mathematically equivalent form, rotation matrix, and used by `getOrientation()` function to compute the orientation in its angular form.

Source: https://android.googlesource.com/platform/frameworks/base/+/b267554/core/java/android/hardware/SensorManager.java

EXHIBIT B

U.S. Patent No. 8,552,978

LG V20

SUBJECT TO CHANGE

U.S. Patent No. 8,552,978 – LG V20

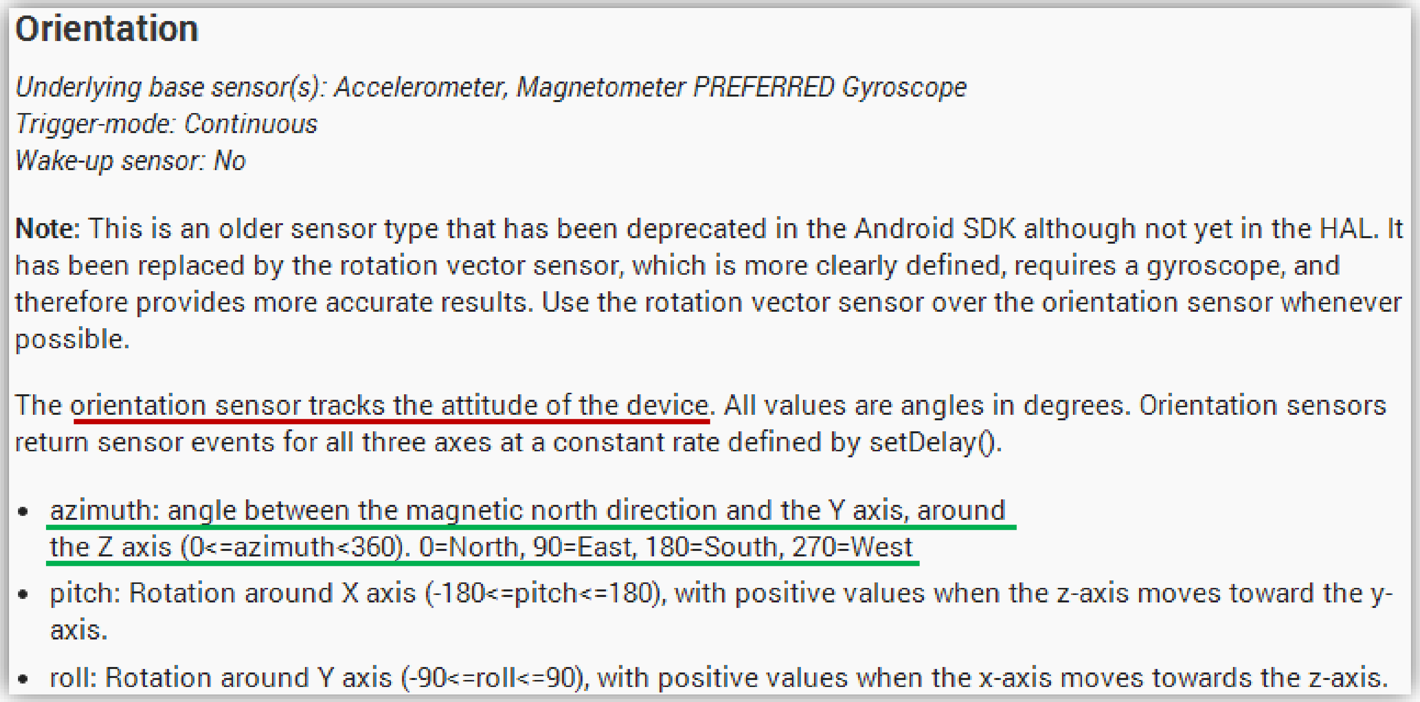
Claim 10
A method for compensating rotations of a 3D pointing device, comprising:
 LG V20

SUBJECT TO CHANGE

Claim 10

generating an **orientation output** associated with an orientation of the 3D pointing device associated with three coordinate axes of a **global reference frame associated with Earth**;

When the orientation sensor is software-based, the **orientation output** is the attitude of the device that can be represented by the azimuth, pitch, and roll angles relative to the magnetic North Pole associated with a **global reference frame associated with Earth**.



Orientation

Underlying base sensor(s): Accelerometer, Magnetometer PREFERRED Gyroscope
Trigger-mode: Continuous
Wake-up sensor: No

Note: This is an older sensor type that has been deprecated in the Android SDK although not yet in the HAL. It has been replaced by the rotation vector sensor, which is more clearly defined, requires a gyroscope, and therefore provides more accurate results. Use the rotation vector sensor over the orientation sensor whenever possible.

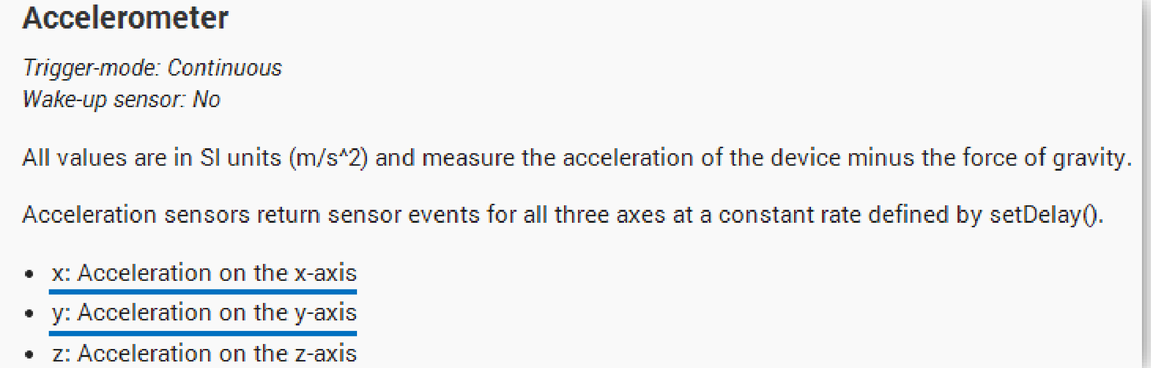
The orientation sensor tracks the attitude of the device. All values are angles in degrees. Orientation sensors return sensor events for all three axes at a constant rate defined by setDelay().

- azimuth: angle between the magnetic north direction and the Y axis, around the Z axis (0<=azimuth<360). 0=North, 90=East, 180=South, 270=West
- pitch: Rotation around X axis (-180<=pitch<=180), with positive values when the z-axis moves toward the y-axis.
- roll: Rotation around Y axis (-90<=roll<=90), with positive values when the x-axis moves towards the z-axis.

Source: https://source.android.com/devices/sensors/composite_sensors.html

Claim 10

generating a **first signal set** comprising axial accelerations associated with movements and rotations of the 3D pointing device in the **spatial reference frame**;



Accelerometer

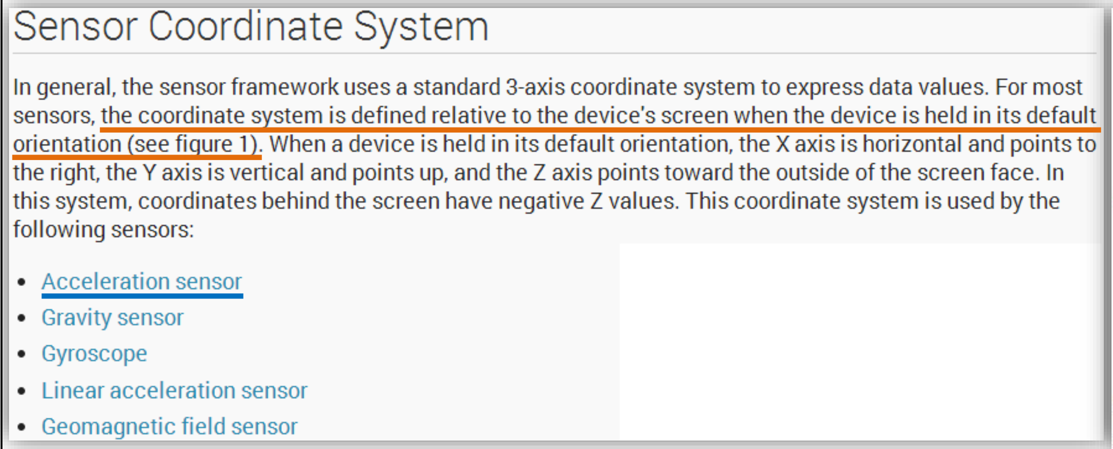
Trigger-mode: Continuous
Wake-up sensor: No

All values are in SI units (m/s^2) and measure the acceleration of the device minus the force of gravity.

Acceleration sensors return sensor events for all three axes at a constant rate defined by setDelay().

- x: Acceleration on the x-axis
- y: Acceleration on the y-axis
- z: Acceleration on the z-axis

Source: https://source.android.com/devices/sensors/base_triggers.html



Sensor Coordinate System

In general, the sensor framework uses a standard 3-axis coordinate system to express data values. For most sensors, the coordinate system is defined relative to the device's screen when the device is held in its default orientation (see figure 1). When a device is held in its default orientation, the X axis is horizontal and points to the right, the Y axis is vertical and points up, and the Z axis points toward the outside of the screen face. In this system, coordinates behind the screen have negative Z values. This coordinate system is used by the following sensors:

- Acceleration sensor
- Gravity sensor
- Gyroscope
- Linear acceleration sensor
- Geomagnetic field sensor

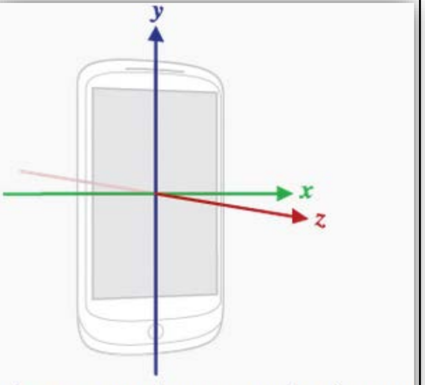
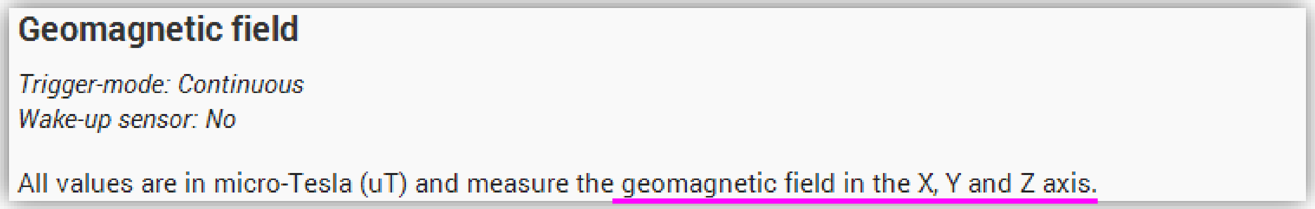


Figure 1. Coordinate system (relative to a device) that's used by the Sensor API.

Source: http://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords

Claim 10
generating a **second signal set** associated with **Earth's magnetism**;
The magnetometer (i.e., the compass) generates a second signal set associated with **Earth's magnetism**.



Geomagnetic field

Trigger-mode: Continuous
Wake-up sensor: No

All values are in micro-Tesla (uT) and measure the geomagnetic field in the X, Y and Z axis.

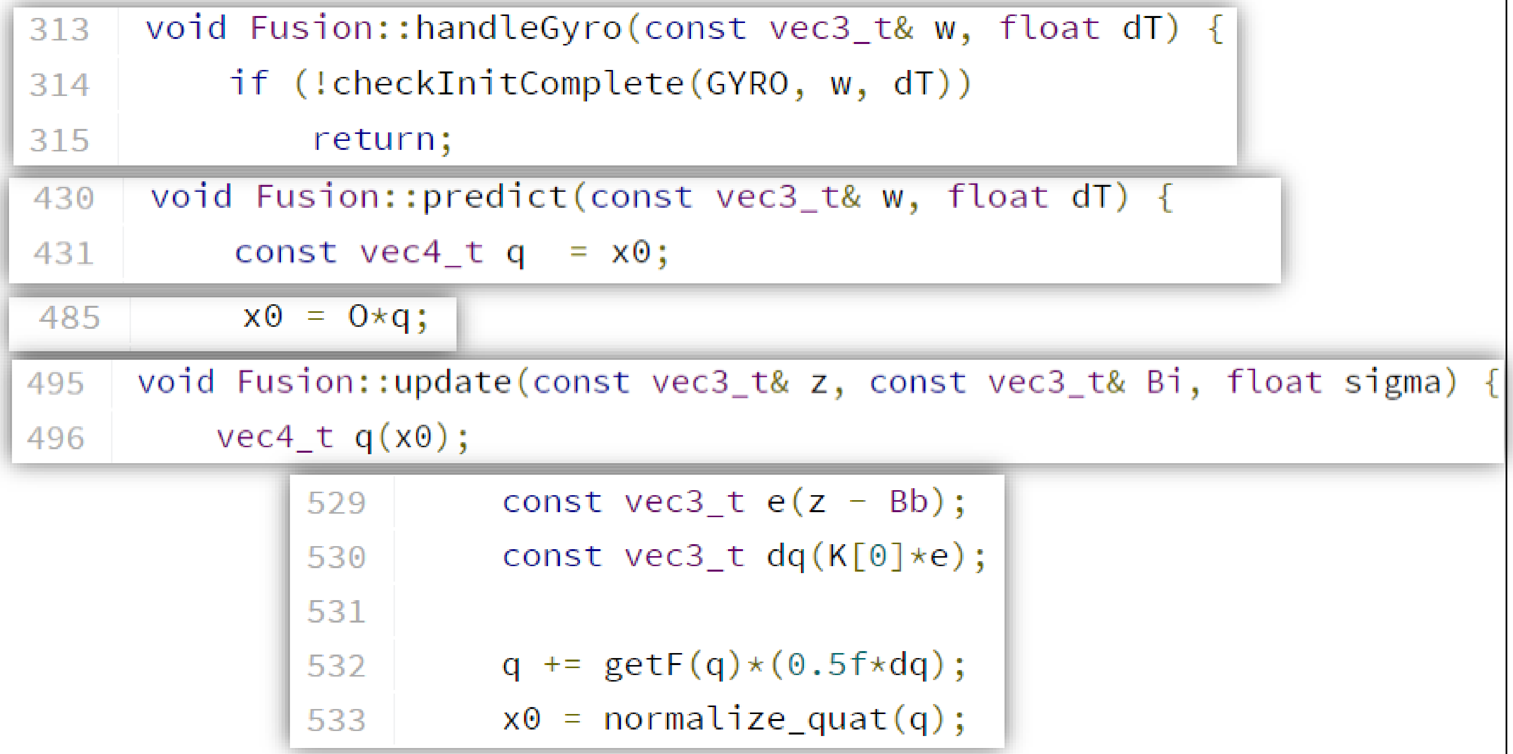
Source: https://source.android.com/devices/sensors/composite_sensors.html

Claim 10

generating the **orientation output** based on the **first signal set**, the **second signal set** and the **rotation output** or based on the **first signal set** and the **second signal set**;

The Android source code shows generating the **orientation output** based on the **first signal set**, the **second signal set** and the **rotation output**.

The `handleGyro()` function passes **rotation output** `w` to the `predict()` function and the `update()` function to calculate an **orientation output**, `x0`.



```
void Fusion::handleGyro(const vec3_t& w, float dT) {
    if (!checkInitComplete(GYRO, w, dT))
        return;
```

```
void Fusion::predict(const vec3_t& w, float dT) {
    const vec4_t q  = x0;
```

```
    x0 = O*q;
```

```
void Fusion::update(const vec3_t& z, const vec3_t& Bi, float sigma) {
    vec4_t q(x0);
```

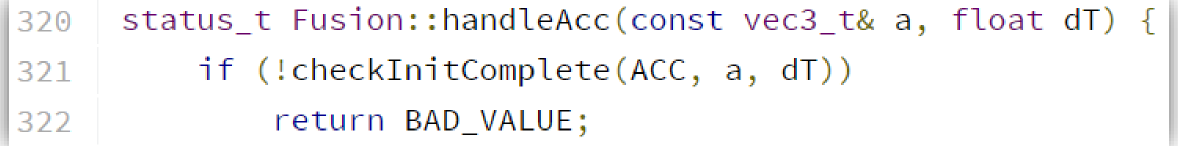
```
    const vec3_t e(z - Bb);
    const vec3_t dq(K[0]*e);

    q += getF(q)*(0.5f*dq);
    x0 = normalize_quat(q);
```

Source: https://android.googlesource.com/platform/frameworks/native/+/master/services/sensorservice/Fusion.cpp

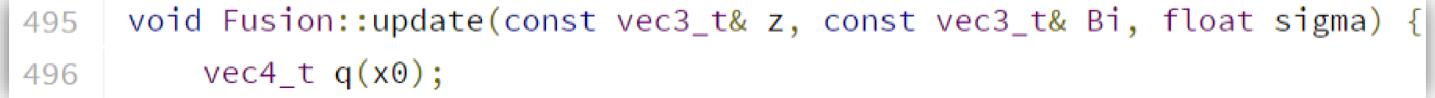
Claim 10

generating the **orientation output** based on the **first signal set**, the second signal set and the rotation output or based on the **first signal set** and the second signal set;

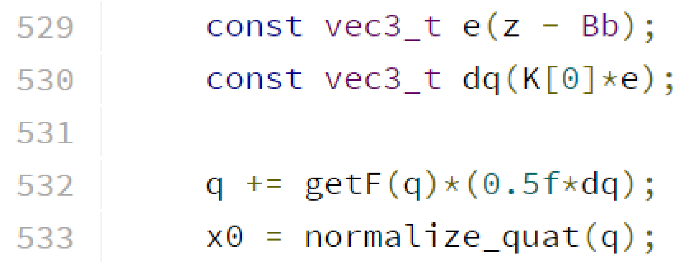
The `handleAcc()` function passes the accelerometer measurements (**first signal set**) `a` to the `update()` function, which updates the **orientation output** `x0`.



```
status_t Fusion::handleAcc(const vec3_t& a, float dT) {
    if (!checkInitComplete(ACC, a, dT))
        return BAD_VALUE;
```



```
void Fusion::update(const vec3_t& z, const vec3_t& Bi, float sigma) {
    vec4_t q(x0);
```



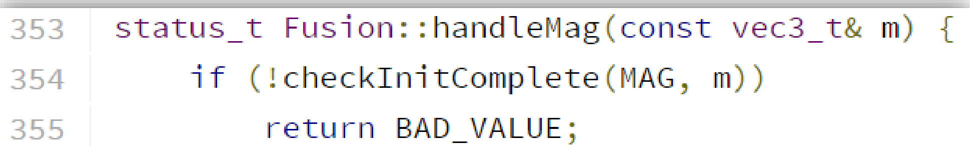
```
    const vec3_t e(z - Bb);
    const vec3_t dq(K[0]*e);

    q += getF(q)*(0.5f*dq);
    x0 = normalize_quat(q);
```

Source: https://android.googlesource.com/platform/frameworks/native/+/master/services/sensorservice/Fusion.cpp

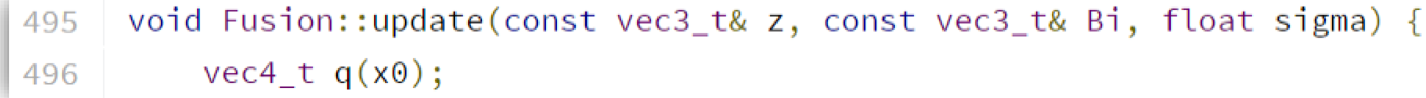
Claim 10

generating the **orientation output** based on the first signal set, the **second signal set** and the rotation output or based on the first signal set and the **second signal set**;

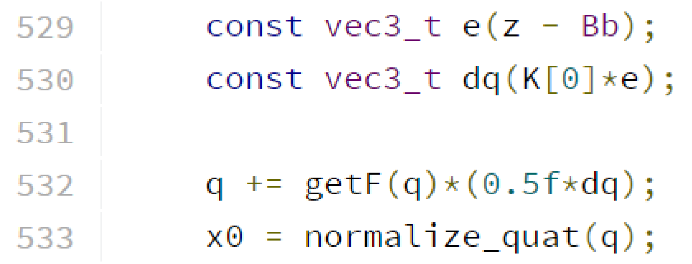
The `handleMag()` function passes the magnetometer measurements (**second signal set**) `m` to the same `update()`, which also updates the **orientation output** `x0`.



```
status_t Fusion::handleMag(const vec3_t& m) {
    if (!checkInitComplete(MAG, m))
        return BAD_VALUE;
```



```
void Fusion::update(const vec3_t& z, const vec3_t& Bi, float sigma) {
    vec4_t q(x0);
```

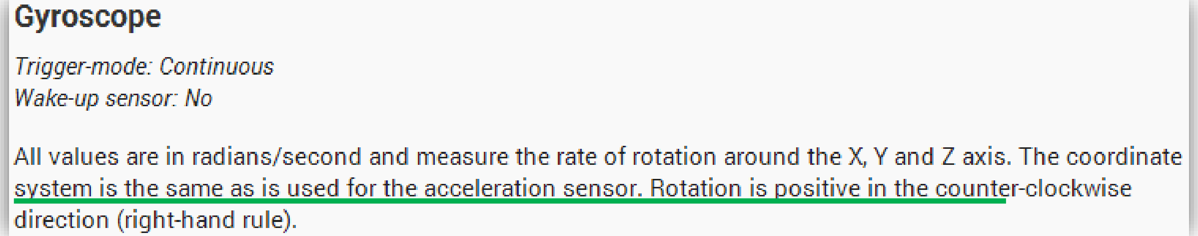


```
        const vec3_t e(z - Bb);
        const vec3_t dq(K[0]*e);

        q += getF(q)*(0.5f*dq);
        x0 = normalize_quat(q);
```

Source: https://android.googlesource.com/platform/frameworks/native/+/master/services/sensorservice/Fusion.cpp

Claim 10

generating a **rotation output** associated with a rotation of the 3D pointing device associated with three coordinate axes of a **spatial reference frame** associated with the 3D pointing device; and

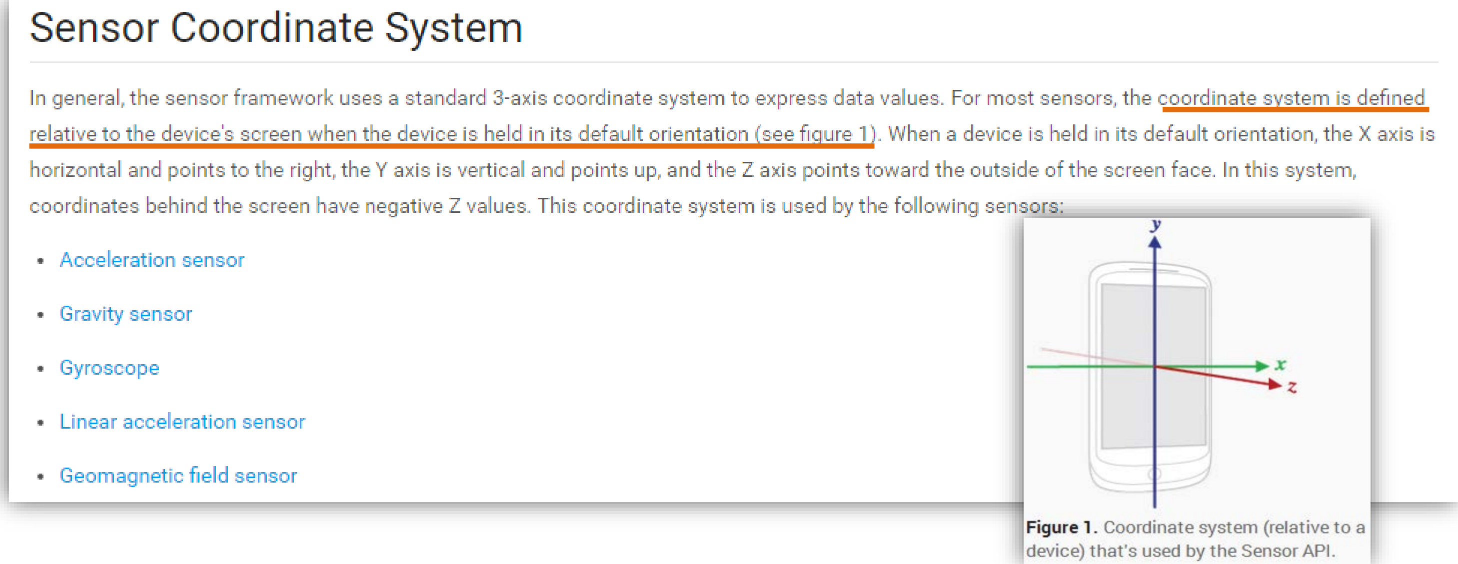


Gyroscope

Trigger-mode: Continuous
Wake-up sensor: No

All values are in radians/second and measure the rate of rotation around the X, Y and Z axis. The coordinate system is the same as is used for the acceleration sensor. Rotation is positive in the counter-clockwise direction (right-hand rule).

Source: https://source.android.com/devices/sensors/base_triggers.html



Sensor Coordinate System

In general, the sensor framework uses a standard 3-axis coordinate system to express data values. For most sensors, the coordinate system is defined relative to the device's screen when the device is held in its default orientation (see figure 1). When a device is held in its default orientation, the X axis is horizontal and points to the right, the Y axis is vertical and points up, and the Z axis points toward the outside of the screen face. In this system, coordinates behind the screen have negative Z values. This coordinate system is used by the following sensors:

- Acceleration sensor
- Gravity sensor
- Gyroscope
- Linear acceleration sensor
- Geomagnetic field sensor

Figure 1. Coordinate system (relative to a device) that's used by the Sensor API.

Source: http://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords

U.S. Patent No. 8,552,978 – LG V20

Claim 10
using the orientation output and the rotation output to generate a transformed output associated with a **fixed reference frame** associated with a **display device**,
The **fixed reference frame** is defined by the horizontal and vertical axes of pixels on LG V20 Edge's **display device**.

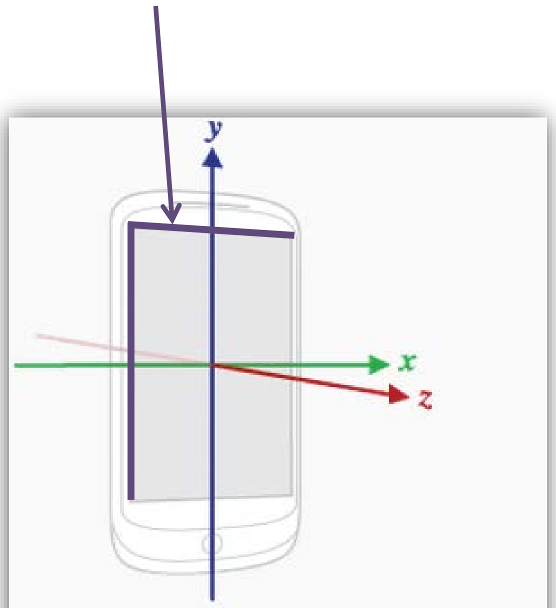


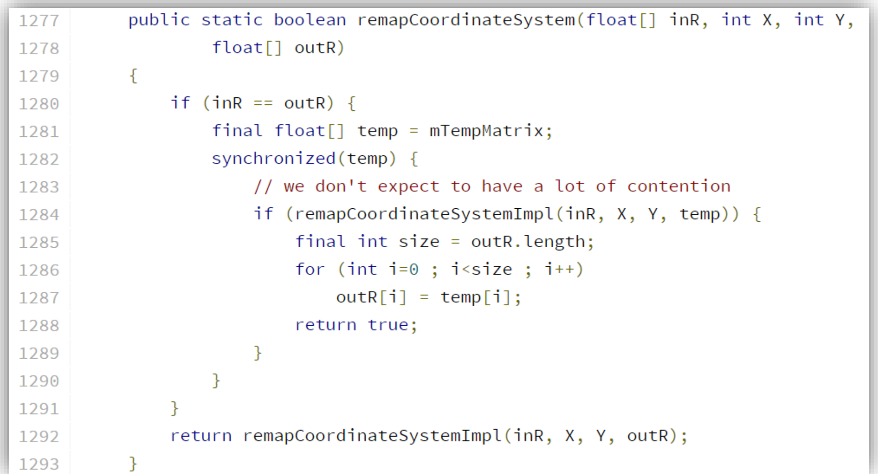
Figure 1. Coordinate system (relative to a device) that's used by the Sensor API.

Source: http://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords

Claim 10

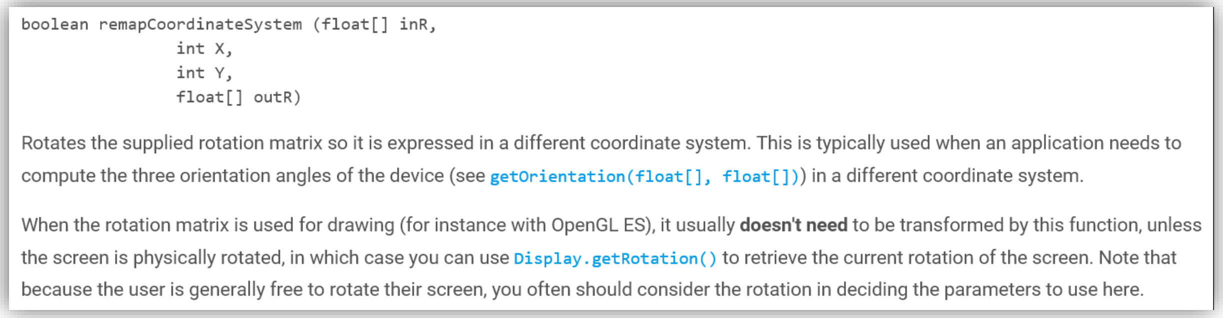
using the orientation output and the rotation output to generate a transformed output associated with a **fixed reference frame** associated with a **display device**,

The `remapCoordinateSystem()` function transforms the orientation output (`inR`) to a transformed output (`outR`), associated with a two dimensional movement in a plane that is parallel to the screen of a **display device**.



```
    public static boolean remapCoordinateSystem(float[] inR, int X, int Y,
            float[] outR)
    {
        if (inR == outR) {
            final float[] temp = mTempMatrix;
            synchronized(temp) {
                // we don't expect to have a lot of contention
                if (remapCoordinateSystemImpl(inR, X, Y, temp)) {
                    final int size = outR.length;
                    for (int i=0 ; i<size ; i++)
                        outR[i] = temp[i];
                    return true;
                }
            }
        }
        return remapCoordinateSystemImpl(inR, X, Y, outR);
    }
```

Source: https://android.googlesource.com/platform/frameworks/base/+/master/core/java/android/hardware/SensorManager.java



```
boolean remapCoordinateSystem (float[] inR,
                int X,
                int Y,
                float[] outR)
```

Rotates the supplied rotation matrix so it is expressed in a different coordinate system. This is typically used when an application needs to compute the three orientation angles of the device (see `getOrientation(float[], float[])`) in a different coordinate system.

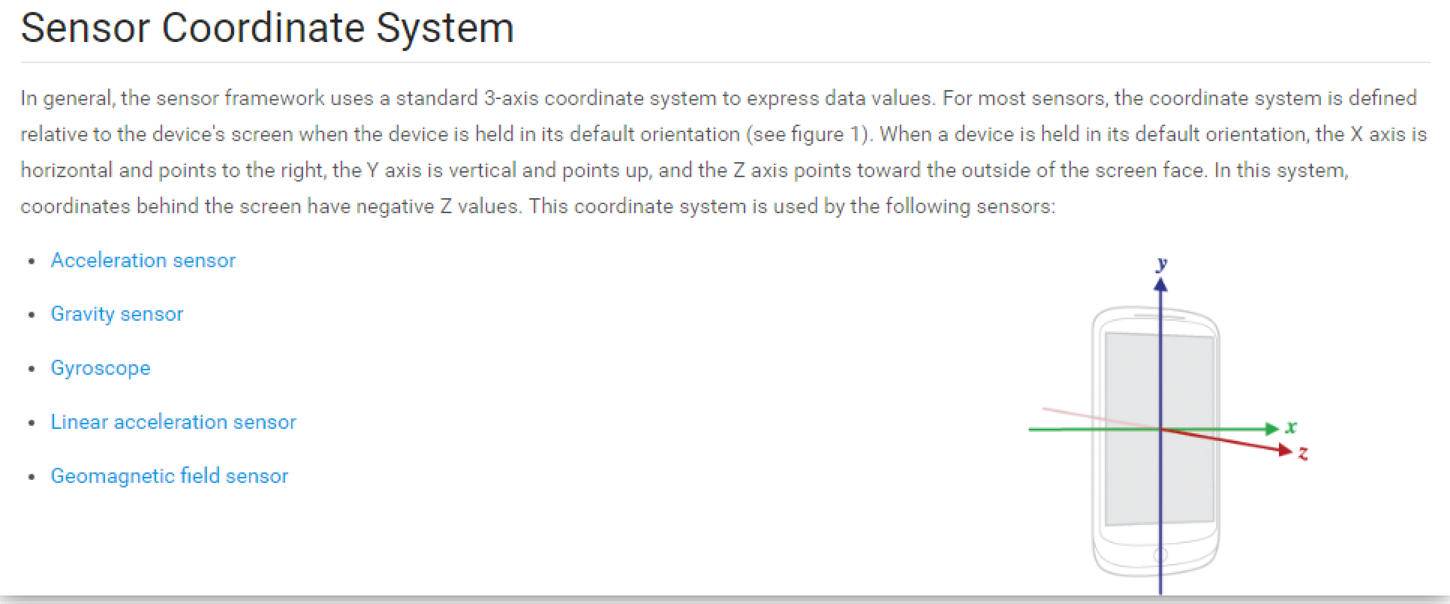
When the rotation matrix is used for drawing (for instance with OpenGL ES), it usually **doesn't need** to be transformed by this function, unless the screen is physically rotated, in which case you can use `Display.getRotation()` to retrieve the current rotation of the screen. Note that because the user is generally free to rotate their screen, you often should consider the rotation in deciding the parameters to use here.

Source: http://developer.android.com/reference/android/hardware/SensorManager.html

Claim 10

wherein the orientation output and the rotation output is generated by a **nine-axis motion sensor module**;

The LG V20 includes a 3-axis gyroscope, a 3-axis accelerometer, and a 3-axis magnetometer which form a **nine-axis motion sensor module.**



Sensor Coordinate System

In general, the sensor framework uses a standard 3-axis coordinate system to express data values. For most sensors, the coordinate system is defined relative to the device's screen when the device is held in its default orientation (see figure 1). When a device is held in its default orientation, the X axis is horizontal and points to the right, the Y axis is vertical and points up, and the Z axis points toward the outside of the screen face. In this system, coordinates behind the screen have negative Z values. This coordinate system is used by the following sensors:

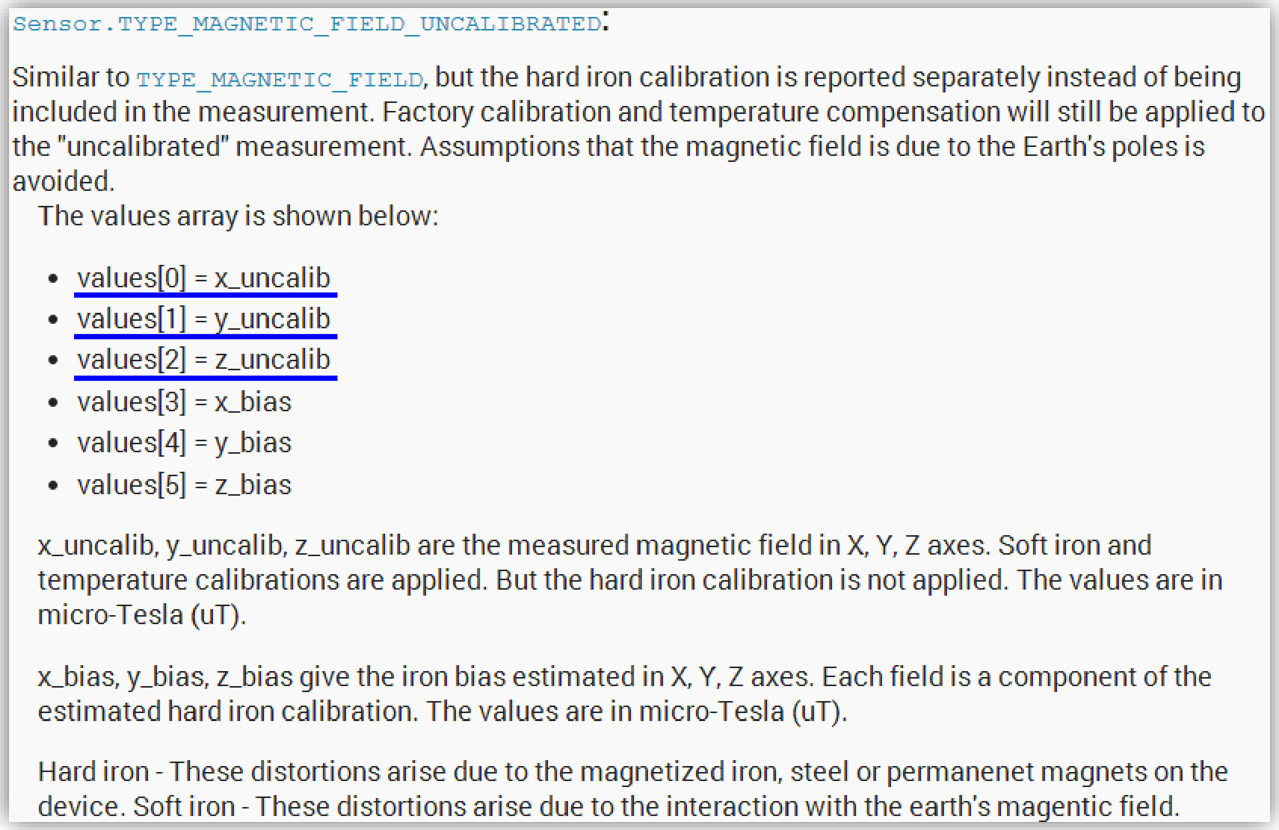
- Acceleration sensor
- Gravity sensor
- Gyroscope
- Linear acceleration sensor
- Geomagnetic field sensor

Source: http://developer.android.com/guide/topics/sensors/sensors_overview.html#sensors-coords

Claim 10

obtaining one or more resultant deviation including a plurality of deviation angles using a plurality of **measured magnetisms Mx, My, Mz** and a plurality of predicted magnetism Mx′, My′ and Mz′ for the second signal set.

The **measured magnetisms Mx, My, Mz** are `values[0]`-`[2]`.



`Sensor.TYPE_MAGNETIC_FIELD_UNCALIBRATED`:

Similar to `TYPE_MAGNETIC_FIELD`, but the hard iron calibration is reported separately instead of being included in the measurement. Factory calibration and temperature compensation will still be applied to the "uncalibrated" measurement. Assumptions that the magnetic field is due to the Earth's poles is avoided.

The values array is shown below:

- values[0] = x_uncalib
- values[1] = y_uncalib
- values[2] = z_uncalib
- values[3] = x_bias
- values[4] = y_bias
- values[5] = z_bias

x_uncalib, y_uncalib, z_uncalib are the measured magnetic field in X, Y, Z axes. Soft iron and temperature calibrations are applied. But the hard iron calibration is not applied. The values are in micro-Tesla (uT).

x_bias, y_bias, z_bias give the iron bias estimated in X, Y, Z axes. Each field is a component of the estimated hard iron calibration. The values are in micro-Tesla (uT).

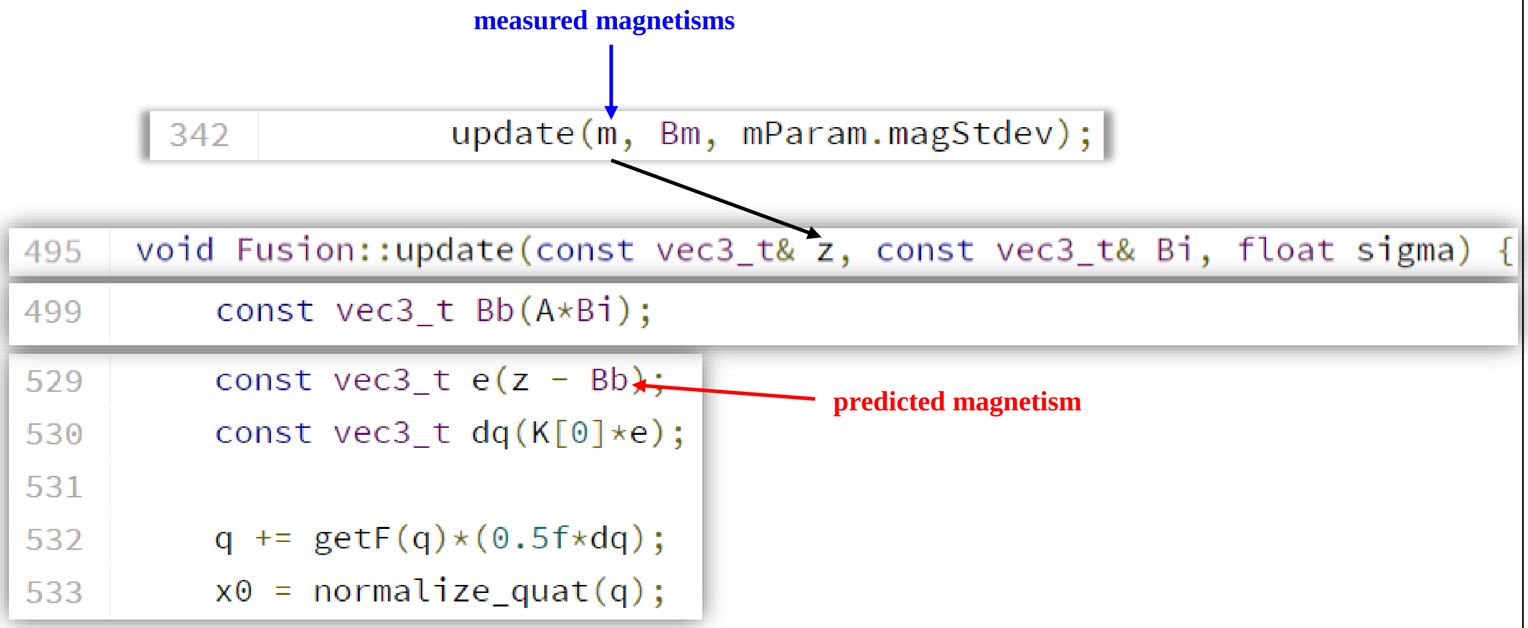
Hard iron - These distortions arise due to the magnetized iron, steel or permanenet magnets on the device. Soft iron - These distortions arise due to the interaction with the earth's magentic field.

Source: http://developer.android.com/reference/android/hardware/SensorEvent.html#values

Claim 10

obtaining one or more resultant deviation including a plurality of deviation angles using a plurality of **measured magnetisms Mx, My, Mz** and a plurality of **predicted magnetism** Mx′, My′ and Mz′ for the second signal set.

The **measured magnetisms**, z, and a **predicted magnetism**, Bb, are used to calculate a global variable x0 in quaternion form.



```
            update(m, Bm, mParam.magStdev);
```

```
void Fusion::update(const vec3_t& z, const vec3_t& Bi, float sigma) {
    const vec3_t Bb(A*Bi);
```

```
    const vec3_t e(z - Bb);
    const vec3_t dq(K[0]*e);

    q += getF(q)*(0.5f*dq);
    x0 = normalize_quat(q);
```

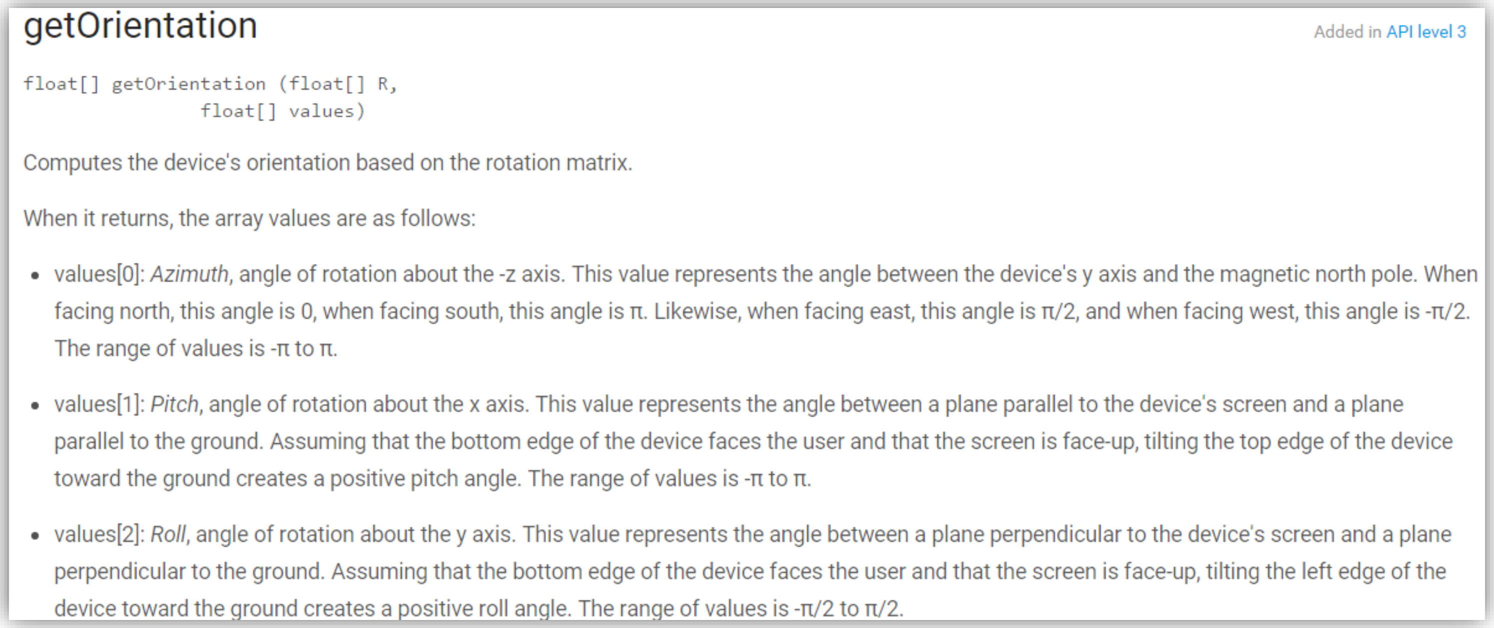
Source: https://android.googlesource.com/platform/frameworks/native/+/master/services/sensorservice/Fusion.cpp

Claim 10

obtaining one or more **resultant deviation including a plurality of deviation angles** using a plurality of measured magnetisms Mx, My, Mz and a plurality of predicted magnetism Mx′, My′ and Mz′ for the second signal set. .

The global variable `x0` is in quaternion form, and can easily be converted to resultant angles.

According to Android’s developer library, the `getOrientation()` function “computes the device’s orientation based on the rotation matrix,” and returns **deviation angles** including the Azimuth, Pitch, and Roll angles.



getOrientation

```
float[] getOrientation (float[] R,
                float[] values)
```

Computes the device's orientation based on the rotation matrix.

When it returns, the array values are as follows:

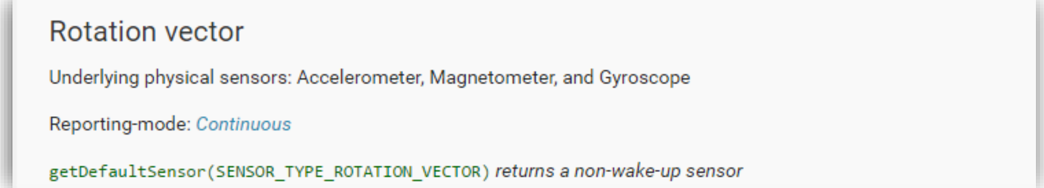
- values[0]: *Azimuth*, angle of rotation about the -z axis. This value represents the angle between the device's y axis and the magnetic north pole. When facing north, this angle is 0, when facing south, this angle is π. Likewise, when facing east, this angle is π/2, and when facing west, this angle is -π/2. The range of values is -π to π.
- values[1]: *Pitch*, angle of rotation about the x axis. This value represents the angle between a plane parallel to the device's screen and a plane parallel to the ground. Assuming that the bottom edge of the device faces the user and that the screen is face-up, tilting the top edge of the device toward the ground creates a positive pitch angle. The range of values is -π to π.
- values[2]: *Roll*, angle of rotation about the y axis. This value represents the angle between a plane perpendicular to the device's screen and a plane perpendicular to the ground. Assuming that the bottom edge of the device faces the user and that the screen is face-up, tilting the left edge of the device toward the ground creates a positive roll angle. The range of values is -π/2 to π/2.

The `getRotationMatrixFromVector()` function “convert[s] a rotation vector to a rotation matrix,” and the `getQuaternionFromVector()` function “convert[s] a rotation vector to a normalized quaternion.” Therefore, the quaternion, `x0`, can be easily converted to its mathematically equivalent form, rotation matrix, and used by `getOrientation()` function to compute the orientation in its angular form.

Source: https://android.googlesource.com/platform/frameworks/base/+/b267554/core/java/android/hardware/SensorManager.java

Claim 10

obtaining one or more **resultant deviation including a plurality of deviation angles** using a plurality of measured magnetisms Mx, My, Mz and a plurality of predicted magnetism Mx′, My′ and Mz′ for the second signal set. .

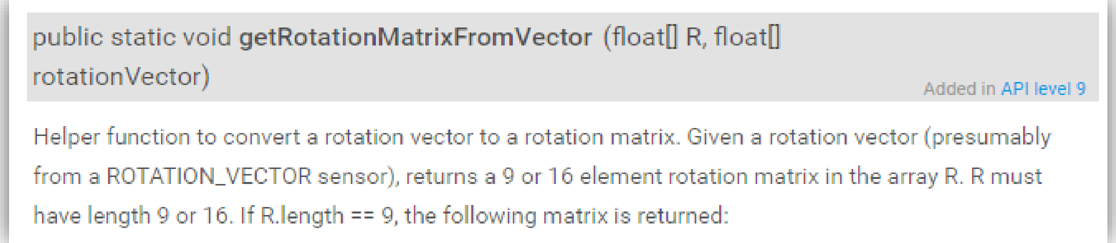


Rotation vector

Underlying physical sensors: Accelerometer, Magnetometer, and Gyroscope

Reporting-mode: *Continuous*

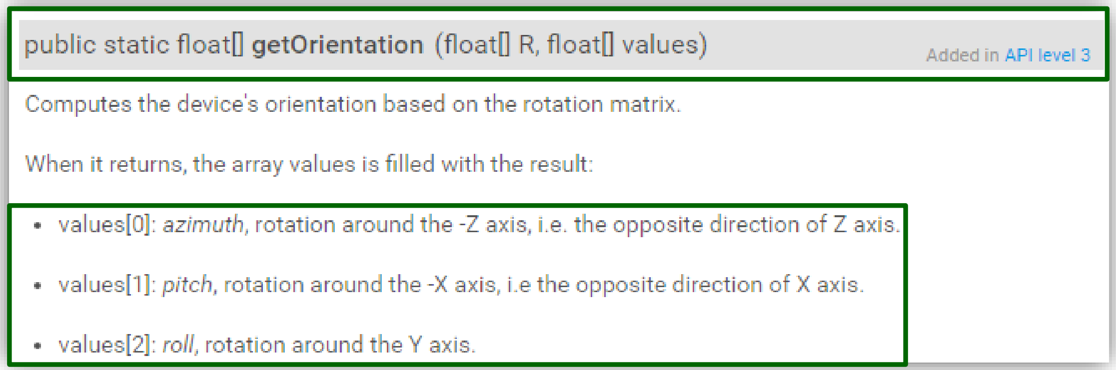
`getDefaultSensor(SENSOR_TYPE_ROTATION_VECTOR)` *returns a non-wake-up sensor*



public static void **getRotationMatrixFromVector** (float[] R, float[] rotationVector)

Added in API level 9

Helper function to convert a rotation vector to a rotation matrix. Given a rotation vector (presumably from a ROTATION_VECTOR sensor), returns a 9 or 16 element rotation matrix in the array R. R must have length 9 or 16. If R.length == 9, the following matrix is returned:



public static float[] **getOrientation** (float[] R, float[] values)

Added in API level 3

Computes the device's orientation based on the rotation matrix.

When it returns, the array values is filled with the result:

- values[0]: *azimuth*, rotation around the -Z axis, i.e. the opposite direction of Z axis.
- values[1]: *pitch*, rotation around the -X axis, i.e the opposite direction of X axis.
- values[2]: *roll*, rotation around the Y axis.

Source: https://developer.android.com/reference/android/hardware/SensorManager.html#getOrientation(float[], float[])

EXHIBIT C

Jill F. Kopeikin (State Bar No. 160792)
Valerie M. Wagner (State Bar No. 173146)
GCA LAW PARTNERS LLP
2570 W. El Camino Real, Suite 400
Mountain View, CA 94040
Telephone: (650) 428-3900
Fax: (650) 428-3901
jkopeikin@gcalaw.com
vwagner@gcalaw.com

(Additional Counsel Identified On Signature Page)
Attorneys for Plaintiff CYWEE GROUP LTD.

IN THE UNITED STATES DISTRICT COURT
FOR THE SOUTHERN DISTRICT OF CALIFORNIA

CYWEE GROUP LTD.,
Plaintiff,

LG ELECTRONICS, INC.,
LG ELECTRONICS U.S.A., INC.,
AND LG ELECTRONICS
MOBILECOMM U.S.A., INC.,
Defendants.

CASE NO. 3:17-cv-01102-BEN-RBB

DECLARATION OF NICHOLAS GANS, PH.D.

DEMAND FOR JURY TRIAL

I, Nicholas Gans, Ph.D., hereby declare as follows:

1. I have been asked by counsel for Plaintiff CyWee Group Ltd. ("CyWee") to offer certain opinions regarding the technologies disclosed in U.S. Patent No. 8,441,438 (the "'438 patent") and U.S. Patent No. 8,552,978 (the "'978 patent").

2. In connection with the preparation of this Declaration, I have reviewed the materials listed below:

- The '438 patent;
- The file wrapper for the '438 patent;
- The '978 patent; and
- The file wrapper for the '978 patent.

3. All of the opinions stated in this declaration are based on my personal knowledge and professional judgment. If called as a witness, I am prepared to testify competently about them.

I. EXPERIENCE AND QUALIFICATIONS

4. I am a Clinical Associate Professor with the Department of Electrical and Computer Engineering at the University of Texas at Dallas.

5. I received my doctorate in Systems and Entrepreneurial Engineering from the University of Illinois at Urbana-Champaign, with dissertation research in the fields of robotics, controls, and estimation. I continue to research and teach these topics in my capacity as a Professor, with over 100 peer-reviewed publications and three patents. I have authored multiple papers on the topic of Inertial Measurement Units and related sensors and fusion algorithms.

6. A more complete list of my qualifications is set forth in my curriculum vitae, a copy of which is attached hereto as Exhibit 1.

7. I am being compensated for work in this matter. My compensation in no way depends on the outcome of this litigation, nor do I have a personal interest in

the outcome of this litigation.

II. NATURE OF THE DISCLOSED TECHNOLOGIES

8. The ’438 patent and ’978 patent are each directed to devices and methods for tracking the motion of a portable electronic device in 3D space and compensating for accumulated errors to map the 3D movements of the device onto a display frame (’438 patent) or transform the 3D movements for a display, such as a 2D display on a computer or handheld device (’978 patent). ’438 patent 1:17-52, 3:52-57; ’978 patent 1:22-27, 7:5-18. That is, at a high level, the patented inventions teach how to determine a device’s current orientation based on motion data detected by its motion sensors, such as an accelerometer, gyroscope, and magnetometer. ’438 patent 4:6-30; ’978 patent 4:15-44. The ’438 and ’978 patents describe portable electronic devices or pointing devices such as smartphones and navigation equipment. ’978 patent 22:34-40, Fig. 6; ’438 patent 4:6-30, Fig. 6.

9. There are different types of motion sensors, including accelerometers, gyroscopes, and magnetometers. Accelerometers measure accelerations. For example, airbags use accelerometers, such that the airbag is triggered based on sudden deceleration. Accelerometers can also measure forces due to gravity. Gyroscopes measure rotation rates or angular velocities. Magnetometers measure magnetism, including the strength of a magnetic field along a particular direction. Each type of motion sensor is subject to inaccuracies. For example, a gyroscope sensor has a small, added offset or bias. This bias will accumulate over time and lead to large drift error. Similarly, magnetometers are subject to interference from natural and manmade sources (*e.g.*, power electronics). Additionally, errors can accumulate over time. These sensors typically take measurements along a single direction. To accurately measure motions along an arbitrary axis, three like sensors are grouped together and aligned at right angles. Such a sensor set is generally referred to as a 3-axis sensor.

10. To incorporate the data from multiple sensors and compensate for the errors described above, the ’438 patent and ’978 patent each disclose a sensor fusion technology. Specifically, the ’438 patent discloses an enhanced 6-axis sensor fusion technology and application for calculating orientation (including tilting angles along all three spatial axes) by using measurements from both a 3-axis accelerometer and a 3-axis gyroscope; furthermore, it can correct or eliminate errors associated with the motion sensors. ’438 patent 4:6-30. This technology is especially suited for accurately representing a mobile device’s orientation in 3D space on a 2D display screen by mapping the yaw, pitch, and roll angles relating to movement along the three spatial axes to a display reference frame, such as that of a 2D display. Simply put, the ’438 patent discloses an improved system and method to capture motion of the device and for eliminating or correcting errors based on movements and rotations of the device.

11. Likewise, the ’978 patent discloses a similar 9-axis enhanced sensor fusion technology for calculating orientation and transforming the movement of the device to a display. ’978 patent 4:15-44. Unlike the ’438 patent, which discloses and claims using two motion sensors—an accelerometer and gyroscope—the ’978 patent discloses and claims using a third sensor—a magnetometer. *Id.*, Claim 1, 10.

12. Orientation information returned by the claimed inventions of the ’438 and ’978 patents has many uses, particularly for mobile cellular devices, such as navigation, gaming, and augmented/virtual reality applications. Navigation applications can use orientation information to determine the heading of the phone, indicate what direction the user is facing, and automatically orient the map to align with the cardinal directions. Increasing numbers of games and other applications use the motion of the phone to input commands, such as tilting the mobile device like a steering wheel. Augmented and virtual reality applications rely on accurate estimation of the device orientation in order to render graphics and images at the

proper locations on the screen.

III. BACKGROUND OF THE TECHNOLOGIES

13. Prior to 2010, motion sensors had limited applicability to portable electronic devices due to a variety of technological hurdles. For example, different types of acceleration (*e.g.*, linear, centrifugal, gravitational) could not be readily distinguished from one another, and rapid, dynamic, and unexpected movements caused significant errors and inaccuracies. These difficulties were compounded by the miniaturization of the sensors necessary to incorporate them in portable electronic devices. With the development of micro-electromechanical systems, or "MEMS," miniaturized motion sensors could be manufactured and incorporated on a semiconductor chip, but such MEMS sensors had significant limitations.

14. For example, it is impossible for MEMS accelerometers to distinguish different types of acceleration (*e.g.*, linear, centrifugal, gravitational). When a MEMS accelerometer is used to estimate orientation, it must measure force along the direction of gravity (*i.e.*, down), but that gravitational measurement can be "interfused" with other accelerations and forces (*e.g.*, vibration or movement by the person holding the device). Thus, non-gravitational accelerations and forces must be estimated and subtracted from the MEMS accelerometer measurement to yield an accurate result. A MEMS gyroscope is prone to drift, which will accumulate increasing errors over time if not corrected by another sensor or recalibrated. A MEMS magnetometer is highly sensitive to not only the earth's magnetic fields, but other sources of magnetism (*e.g.*, power lines and transformers) and can thereby suffer inaccuracies from environmental sources of interference that vary both in existence and intensity from location to location.

15. Additionally, orientation cannot be accurately calculated using only one type of MEMS sensor. For example, if only a 3-axis MEMS accelerometer is used to measure orientation, pitch and yaw can be measured, but not roll. If only a MEMS

gyroscope is used to measure angular velocity, only relative changes in orientation can be measured, not absolute orientation.

16. Without orientation information, mobile device apps would be limited to very static operation. This was the scenario with initial smart phones and other mobile devices. Navigation aids could render a map and indicate the location of the device using GPS. However, these maps would orient with North on the map pointing to the top of the screen. The user could rotate the map using touch commands, but the map would not rotate automatically as the user turned. Nor could the device indicate what direction the device was facing.

17. Many games use motion of the device to control the game. A common control scheme, especially for driving and piloting games, is to have the user rotate the device, such as a phone or game controller, like a steering wheel to indicate the direction the vehicle should move. Some puzzle games also use motions to cause elements of the game to move. As discussed previously, accelerometers measure acceleration, which is a very noisy signal. Acceleration is the derivative of velocity, which is the derivative of position. Small magnitude noise can have large derivatives, which means that small levels of noise from vibration or electrical fluctuations will be magnified at the acceleration level. Even a stationary device will have notable noise measured by an accelerometer. A moving device will only amplify this noise. Since accelerometers measure linear and centripetal accelerations as well as the acceleration of gravity, orientation estimates on a moving device will not be accurate. In Figure 1[1], we conducted an experiment where an accelerometer was placed on a moving pendulum with a very accurate angle measurement from an optical encoder. When the pendulum is rotated slowly, the accelerometer is fairly

---

[1] T. R. Bennett, R. Jafari and N. Gans, "Motion Based Acceleration Correction for Improved Sensor Orientation Estimates," 2014 11th International Conference on Wearable and Implantable Body Sensor Networks, Zurich, 2014, pp. 109-114.

accurate, but during moderate or fast motion it shows significant errors.

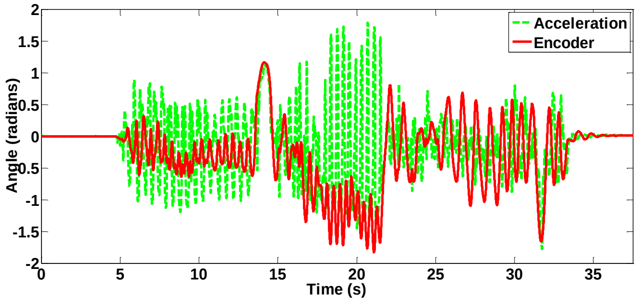


Figure 1

18. If only an accelerometer is used, a coarse estimate of the device orientation can be obtained by averaging or numerically filtering the results. Essentially, the device can determine if it is tilted left or right, up or down, but the exact angle cannot be estimated accurately while in motion. This is suitable for games to move a character or steer a vehicle in a particular direction, but generally cannot utilize the magnitude of tilt to move at corresponding faster or slower speeds.

19. Movement on a display can, of course, be controlled by means other than a portable electronic device with orientation sensors. For example, games could be controlled using traditional “joystick” type inputs. For smart phones with touch screens, commands are given by having the user touch specific parts of the screen.

20. For other current applications, portable electronic devices with orientation sensors are more crucial. Augmented reality (AR) and virtual reality (VR) are new and growing classes of applications for smart phones and mobile devices. In AR, the device camera provides live video feed to the screen, and the application overlays generate graphics onto the screen at specific locations. AR navigation apps can draw signs or labels to indicate what specific places or objects are, or can render arrows or other indicators. AR games and teaching applications can label objects or draw characters or items such that they appear as if they are in the real world seen in the video. Virtual reality is similar but does not use the

camera, rather it completely renders an artificial 3D environment on the screen. VR most often requires a head set such that the user only sees the screen. Mobile devices and smart phones used for VR generally split the screen and display to two side-by-side images of the rendered environment that are slightly offset to simulate a left and right eye. The device then sits in a headset with lenses such that the user has each eye see only one of the split-screen images and has a sense of stereo (3D) vision.

21. Without orientation sensing, AR and VR applications cannot work. The system will have no ability to understand the orientation of the device and know where to draw objects and/or the scene. The rough orientation estimate provided by an accelerometer (ideally with a magnetometer) will not be sufficient to track during typical head motions. It has been demonstrated that VR applications that use an accelerometer often cause motion sickness, as the rendered images do track with the head motions. An AR application with the use of a gyroscope and fusion algorithm will not render objects at the correct locations, and may obscure the view rather than provide helpful information.

22. There are ways to estimate orientation other than the approaches presented in the ’438 patent or ’978 patent, which involve algorithms that filter and fuse measurements from inertial and magnetic sensors. Most such methods are based on cameras and computer vision algorithms. However, the limitations of these methods render them unusable for portable electronic devices. For example, there are a variety of motion capture systems that use cameras arrayed around an environment. Markers (*e.g.*, reflective balls) can be placed on objects, and the cameras can locate the markers, often to sub-mm accuracy. If an object has three or more markers on it, the orientation of the object can be determined with sub-degree accuracy. This method is very accurate, but quite expensive (often about $100,000). The cameras are fixed in place, and the estimation can only work within a small space (a box of dimensions on the order of tens of meters). Clearly, this is not

suitable for the vast majority of mobile device users or applications.

23. A camera on a portable electronic device, such as a smart phone, can be used to estimate orientation of the phone. One class of approaches to this problem uses special patterns or markers in the environment. These often have the appearance of a QR code or 2D UPC. Taking a picture of the pattern, computer vision algorithms can determine the position and orientation of the camera with respect to the marker. AR applications have placed the patterns on specific objects or consumer products so the device can render images and graphics with respect to the pattern. AR games have included patterned mats that are placed on a table or other flat surface, and the device renders characters and objects as if they were on the surface. An example of a game using different patterned mats is seen in Figure 2[2].

Figure 2

24. Multiple unique patterns can be placed around an environment; so long

---

[2] https://pokemondb.net/spinoff/pokedex-3d

(footnote continued)

as one is always in view, the camera can maintain an estimate of the orientation and position. In this way, it can be used for navigation. An example is seen in Figure 3[3]. The necessity of placing patterns would make this approach useless for a majority of applications, particularly outdoors. The camera would also need to remain on at all times, which would cause severe battery drain.

Figure 3

25. Orientation of the camera can also be estimated over an indefinite amount of time using vision algorithms known as visual odometry. In visual odometry, changes in the image over time are used to estimate the camera velocity. This velocity can be integrated over time to estimate the change in orientation. While these methods are well understood, they can only track change in relative orientation, not give absolute orientation. They also require the camera to be on at all times, which will greatly reduce battery life.

IV. OPINION REGARDING ADVANTAGES OVER PRIOR ART

26. As noted in both the ’438 and ’978 patents, prior art portable electronic

---

[3] This is an unpublished image from Dr. Gans’ research laboratory.

devices, such as pointing devices, smartphones and navigation equipment, had several deficiencies in addressing the technological challenges of mapping and transforming movement in a 3D space to a 2D display. These prior art portable devices could only output the movement of the device in 2D, rather than the 3D reference frame of the ’438 and’978 patents. ’438 patent 2:47-55; ’978 patent 2:41-58. In addition, the portable devices could not accurately calculate and account for movements of the device in a dynamic environment, such as erroneous drift measurements of the device or accelerations along with the direction of gravity. ’438 patent 2:55-62; ’978 patent 2:58-66. These prior art portable devices were also limited to detecting gravitational acceleration detected by the accelerometer, and were therefore incapable of accurately outputting the actual yaw, pitch and roll angles. ’438 patent 2:62-3:5; ’978 patent 2:66-3:13. Finally, for the specific case of pointing devices, when they extended beyond the border or boundary of the display, the absolute movement pattern was not mapped, but instead the location outside the boundary was ignored and a relative movement pattern used, which resulted in uncompensated errors. ’438 patent 3:16-51; ’978 patent 3:20-52.

27. The ’438 patent and ’978 patent technologies overcome technological hurdles such as those discussed above and the limitations of the foregoing prior art in a unique and novel way by incorporating measurements from multiple sensors for increased accuracy and algorithms to compensate for accumulated errors. The ’438 patent discloses and claims a system and method for interactively and iteratively fusing angular velocity measurements in the x, y, and z directions and axial acceleration measurements in the x, y, and z directions such that the measurements complement each other according to a specific algorithm. *See, e.g.*, ’438 patent, Claim 1, 14. The limitations of the specific measurements and accumulated errors therein are overcome and eliminated. The ’978 patent builds upon the ’438 patent and further incorporates magnetism measurements in the x, y, and z directions for

increased accuracy. *See, e.g.*, ’978 patent, Claim 1, 10.

28. Accurately and reliably mapping and transforming the orientation of a device in a 3D space to a 2D display presents substantial technical issues. MEMS accelerometers, gyroscopes, and magnetometers all have strengths and weaknesses in their ability to sense data required to calculate orientation on a portable electronic device as described above. The fusion of these data sources (accelerometer and gyroscope in the ’438 patent and accelerometer, gyroscope, and magnetometer in the ’978 patent) and the conversion of relevant data to a practical display in the patented manner is non-trivial, particularly at the time the applications for those patents were filed, and I believe a person of ordinary skill in the art would have been challenged in inventing what the ’438 and ’978 patents claim.

V. OPINION REGARDING PATENTABILITY OF THE ’438 AND ’978 PATENTS

29. I understand that an invention is patentable if the claims, when read in light of the specification, claim patentable subject matter that is new and nonobvious in light of the prior art.

30. I further understand that an invention constitutes patentable subject matter if it is more than an abstract idea or law of nature and/or represents a technological solution to a technological problem. I also understand that an invention is not abstract simply because mathematics is utilized to implement the invention, particularly in the computer arts. Rather, even the use of known components is patentable where, for example, the system utilizes an unconventional approach that provides multiple advantages over the prior art. I understand that the fundamental concern is that a patent claim not be so broad or general in scope that it will preempt all solutions to the problem being addressed by the invention.

31. I also understand that an issued patent is presumed to be valid, that is, it is presumed to be new and nonobvious until proven by clear and convincing

evidence to the contrary.

32. Applying these understandings to the ’438 and ’978 patents, it is my opinion that they claim patentable inventions.

33. The inventions of the ’438 and ’978 patents relate to the highly-complex field of mapping orientation and transforming movement of a portable electronic device in a 3D space to a display, which, as noted above, has a number of technological challenges. The inventions employ a departure from the then-existing prior art, including 6-axis and 9-axis systems that calculate and output absolute movements, not just relative movements using a 3D pointing device, and, in the case of the ’978 patent, the additional use of a magnetometer. ’438 patent 4:6-19, Claim 1, 14; ’978 patent 4:15-32, Claim 1, 10. The inventions present multiple advantages over the prior art, including the elimination and correction of errors in transforming movement in the 3D reference frame to the display. ’438 patent 4:20-30; ’978 patent 4:33-44. While the inventions present an improved device and method for transforming movement in the 3D reference frame to a display, they do not foreclose all other ways of performing the same task.

34. In short, the ’438 and ’978 patents claim a particular configuration of motion sensors incorporated into a portable electronic device and a method of using the data from those sensors to more accurately track and output the movement of the portable electronic device. The mathematical operations are required for the proper operation of the patented systems, but they are a consequence of the arrangement of the sensors and the requirements for the proper operation of the claimed pointing devices and practice of the claimed methods.

35. The inventions of the ’438 and ’978 patents are, therefore, not mere abstractions. Moreover, the inventions represent a technological solution to the challenges in a highly technical area.

36. As to novelty and nonobviousness, I presume that the patents were duly

examined and properly found to be valid over the prior art, including the art that is described and discussed in the respective specifications of the ’438 and ’978 patents. In addition, as noted above, it is my opinion that the inventions of the ’438 and ’978 patents are non-trivial advancements in a complicated field of technology, and that those of skill in the art at the time the applications were filed would have had great difficulty in developing the solutions disclosed and claimed in the patents.

37. Thus, the sensor fusion technology of the ’438 and ’978 patents, including the claimed configuration of sensors and their method of operation, is, in my opinion, patentable.

I declare under penalty of perjury that the foregoing is true and correct to the best of my knowledge and understanding.

Dated: October 6, 2017

Nicholas R Gans

Nicholas Gans, Ph.D.

EXHIBIT 1

Nicholas R. Gans
Curriculum Vitae

ngans@utdallas.edu, nrgans@gmail.com
Department of Electrical Engineering, MS: EC33
The University of Texas at Dallas
800 W. Campbell Rd.
Richardson, TX 75080, USA
http://www.utdallas.edu/~ngans

Phone: (972) 883-6755
Fax: (972) 883-2710

CURRENT POSITION

- **Assistant Professor** – Department of Electrical Engineering, University of Texas at Dallas, Aug. 2009 – present

EDUCATION

- Ph.D. Systems and Entrepreneurial Engineering
  University of Illinois Urbana Champaign - December 2005
  Dissertation: *Hybrid Switched System Visual Servo Control*
  Advisor: Seth Hutchinson
- M.S. Electrical and Computer Engineering
  University of Illinois Urbana Champaign - May 2002
  Thesis: Performance Tests of Partitioned Approaches To Visual Servo Control
  Advisor: Seth Hutchinson
- B.S., *Cum Laude*, Electrical Engineering and Applied Physics, Minor in Philosophy
  Case Western Reserve University - May 1999

RESEARCH INTERESTS

- Vision-Based Control and State Estimation
- Mobile Robot Control
- Multi-Agent/Distributed Control and Estimation
- Self-Optimizing Systems
- Nonlinear Control
- Hybrid Switched-System Control
- Computer Vision
- Human/Machine Interaction
- Engineering Ethics Education

RESEARCH EXPERIENCE

- **Postdoctoral Associate** – National Research Council/Air Force Research Laboratory
  Eglin Air Force Base, Sep. 2008 – July 2009
  Directed by David Jeffcoat
- **Postdoctoral Researcher** - Nonlinear Controls and Robotics Lab
  University of Florida, Jan. 2006 – Aug. 2008
  Directed by Dr. Warren Dixon
- **Graduate Research Assistant** - Robot Motion Planning and Control Lab
  University of Illinois Urbana Champaign, Jan. 2002 - Dec. 2005
  Directed by Dr. Seth Hutchinson

EXHIBIT 1
PAGE 40

CURRENT GRANTS

- PI, "Distributed Control and Vision-Based Estimation for UAS Autonomy," Air Force Research Laboratory Munitions Directorate, 5/01/2017 – 4/31/2019, $75,000
- PI, "Engineering Projects in Community Service," UT Dallas Center for Teaching and Learning Instructional Improvement Grant, 9/1/2016-8/31/2017, $5000
- PI, "Vision-Based Control of Quadrotor UAVs", Kasling Aircraft, 9/1/2016-12/1/2016, $8,098
- PI, "GOALI: Adaptive Control of Inkjet Printing on 3D Curved Surfaces", National Science Foundation, 5/1/2016 – 4/30/2019, $294,452
- PI, "Vision-Based Surface Shape & Condition Estimation", Texas Instruments, 12/1/15 - 11/30/17, $70,000
- PI, "ICC for Olalekan Ogunmolu" (Soft for RadioSurgery), UT Southwestern Medical Center, 9/1/2014 – 5/31/2016, $13,320
- Co-PI, "Engineering Ethics as an Expert Guided and Socially Situated Activity," NSF, 9/1/2013-8/31/2017, $299,967

PREVIOUS GRANTS

- PI, "Smart City Transportation System: Real-Time Multi-Objective Optimization of Autonomous Multi-Agent Controllers in an Orchestrated Resource Environment for Adaptive and Responsive Traffic Management," Texas Research Alliance, 10/1/2015-6/1/2016, $40,000
- PI, "Human Machine Interface for Including Out-of Sequence Information in UAV Target Search," Air Force Research Laboratory, 9/23/2013 – 5/23/2016, $159,998
- Co-PI, "Cooperative Robot Manipulation", Daegu Gyeongbuk Institute of Science And Technology, 3/1/2013 12/31/2015, $272,961
- PI, "3D Interactive Camera/Projector Display Unit for Human/Robot Interaction", Texas Instruments, 1/1/13 - 12/28/15, $105,000
- PI, "Development of Algorithms and an Instructional Lab for Actin Robot Control Software and Cyton Robot Arm ," Energid Technologies Corporation, 1/15/13 – 6-15-13, $23,556
- Co-PI, "Development of an Adaptive Radar Laboratory," Mustang Technology Group, $35,000
- PI, "System Characterizations of the Scanning Tunneling Microscope", Zyvex Corporation, 5/30/2011 - 8/30/2011, $3,084
- PI, "Super-resolution of Target Details for Improved Target State Estimation and Classification," University of Texas Catalyst Grant, June 2010-August 2011, $40,000
- PI, "Hardware in the Loop Simulation for Vision-Based Control of Autonomous Vehicles," Air Force Research Lab grant FA8651-10-1-003, January 2010-September 2010, $10,000

TEACHING EXPERIENCE

- **Course Instructor** University of Texas at Dallas
  - Engineering Projects in Community Service – Spring 2016, Fall 2016
    Linear Systems and Signals (EECs 6331) – Spring 2010, Spring 2012, Spring 2013, Spring 2014, Spring 2015, Spring 2016
  - Introduction to Robotics (EEGR 5V80, ENGR 5375) – Spring 2011, Fall 2013, Fall 2014, Fall 2015, Fall 2015

- Senior Design 2 (EE 4389) – Spring 2012, Spring 2013, Spring 2014, Spring 2015, Spring 2016
- Vision-Based Estimation and Control (EESC 7V85) – Fall 2010, Fall 2012
- Electronic Circuits Laboratory (EE 3111) – Summer 2011, Summer 2012
- Systems and Controls (EE 4310) – Fall 2011

- **One Day Workshop** Recent Advances in Extremum Seeking Control and its Applications
  19th IFAC World Congress, Capetown, South Africa, Sep. 2014
- **One Day Workshop** Vision-Based Estimation and Control
  University of Johannesburg, Johannesburg, South Africa, Nov. 2009

- **Short Course -** Vision-Based Control for Autonomous Vehicles
  NASA Johnson Space Center, Houston, TX, Aug. 2012
  AIAA Guidance, Navigation and Control Conference, Chicago, IL, Aug. 2009
  AIAA Guidance, Navigation and Control Conference, Hilton Head, SC, Aug. 2007
  A two day short course offered through AIAA. I taught sections on cameras and imaging, pose estimation through epipolar geometry, visual servoing, chained pose estimation, and hardware in the loop simulation.

STUDENT SUPERVISION

Doctoral advisement/direction:

- Jinglin Shen, PhD awarded August 2013, “Multi-View Systems for Robot-Human Interaction and Object Grasping”
- Yinghua Zhang, PhD awarded May 2014, “Improving Global Properties of Real-Time Optimization With Applications in Robotic Visual Search”
- Jingfu Jin, PhD awarded December 2015, “Unified Formation Control, Heading Consensus and Obstacle Avoidance for Heterogeneous Mobile Robots with Nonholonomic Constraints”
- J.-Pablo Ramirez, PhD awarded May 2016, “Mobile Sensor Guidance for Optimal Information Acquisition Under Out-Of-Sequence and Soft Measurements”
- Terrell Bennet, PhD awarded August 2016, “Algorithms for Enabling Wearable Sensors in the Internet of Things”

Masters advisement/direction:

- Keveh Fathian, MS awarded December 2012, “Virtual Thermal Sensing and Control of Heat Distribution Using State Estimation”
- David Tick, MS awarded May 2011, “Fusion of Discrete and Continuous Epipolar Geometry With Wheel and IMU Odometry for Localization of Mobile Robots”
- Wen Yu, MS awarded August 2011, “Interactive Camera/Projector Display Unit Using Image Homography”

AWARDS

- **Selected as one of 2014’s IEEE Transactions on Robotics Outstanding Reviewers**
- **Nominated for the Provost's Award for Faculty Excellence in Undergraduate Research Mentoring**, The University of Texas at Dallas, 2014

- **Best Paper of the Session**, K. Fathian, J. Jin and N. Gans, "A New Approach for Solving the Five-Point Relative Pose Problem for Vision-Based estimation and Control," *Proc. American Control Conference 2014*
- **Best doctoral colloquium award (work in progress):** Quality Enhancement Framework for Wearable Computers - Terrell R. Bennett, (my PhD student), 2014 Proc. Body Sensor Network Conference
- **Best Paper of the Session**, T. Bennett, R. Jafari, and N. Gans, "An Extended Kalman Filter to Estimate Human Gait Parameters and Walking Distance," 2013 *Proc. American Controls Conference*
- **Best Student Paper,** D. Q. Tick, J. Shen, and N.R. Gans, "Fusion of Discrete and Continuous Epipolar Geometry for Visual Odometry and Localization," 2010 *IEEE International Workshop on Robotic and Sensors Environments*
- National Research Council/US Air Force Office of Scientific Research Associateship
- **Best Paper of the Session**, K. Kaiser, N. Gans and W. E. Dixon, "Localization and Control of an Aerial Vehicle through Chained, Vision-Based Pose Reconstruction," 2007 *American Control Conference*

PROFESSIONAL ACTIVITIES

Memberships

- IEEE Senior Member
- IEEE Robotics and Automation Society
- IEEE Control Systems Society
- AIAA
- ASME

Conference Organizing Committees

- Local Arrangements Chair - IEEE Conference on Automation Science and Engineering (CASE) 2016
- Exhibitions Chair - IEEE Conference on Automation Science and Engineering (CASE) 2016
- Exhibitions CoChair - IEEE Int'l Conf. on Intelligent Robots and Systems (IROS) 2014

International Program Committee Member

- IEEE Int'l Conf. on Intelligent Robots and Systems (IROS): 2006
- IEEE Int'l Conf. on Intelligent Robots and Systems (IROS), Associate Editor: 2007-2014
- IEEE/IFAC Int'l Conf. on Inform. in Control, Autom. & Robotics (ICINCO): 2008-2014
- American Controls Conference (ACC) 2009-2014

Workshop/Invited Session Organizer/Lecturer

- *6º Taller de Robótica y Planificación de Movimientos*, Plenary Speaker, *Decentralized Formation Control and Obstacle Avoidance for Mobile Robots*, Centro de Investigación en Matemáticas, Guanajuato, Mexico, April. 2016
- *19th IFAC World Congress*, Organizer and Lecturer at One Day Workshop, *Recent Advances in Extremum Seeking Control and its Applications*, Capetown, South Africa, Sep. 2014
- *University of Johannesburg*, Organizer and Lecturer at One Day Workshop, *Vision-Based Estimation and Control*, Johannesburg, South Africa, Nov. 2009

- *AIAA Guidance, Navigation and Control Conference* (GNC), Organizer and Lecturer at Two Day Workshop, *Vision-Based Control for Autonomous Vehicles*, Chicago, IL, Aug. 2009
- *International Conference on Pattern Recognition* (ICPR), Presenter at One Day Workshop, *Visual Observation and Analysis of Animal and Insect Behavior*, Tampa, FL, Dec. 2008
- *IEEE International Symposium on Intelligent Control* (ISIC), Invited Session Organizer and Lecturer: *Current Topics Vision-Based Control*, San Antonio, TX, Sep. 2008
- *AIAA Guidance, Navigation and Control Conference* (GNC), Organizer and Lecturer at Two Day Workshop, *Vision-Based Control for Autonomous Vehicles*, Hilton Head, SC, Aug. 2007

Review Panels

- National Science Foundation
- NASA

Publication Reviewer

- IEEE Transactions on Control (TAC)
- IEEE Transactions on Robotics (TRO)
- IEEE Transactions on Systems, Man and Cybernetics (TSMC)
- IEEE Transactions on Controls System Technology (TCST)
- ASME Journal of Dynamic Systems, Measurement and Control
- International Journal of Computer Vision (IJCV)
- International Journal of Robotics Research (IJRR)
- Journal of Intelligent and Robotic Systems
- European Journal of Control (EJC)
- IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
- IEEE Conference on Robotics and Automation (ICRA)
- IEEE Conference on Decision and Control (CDC)
- American Controls Conference (ACC)
- ASME Dynamic Systems and Control Conference (DSCC)
- IEEE Conference on Control, Automation, Robotics and Vision (ICARV)
- IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)
- International Conference on Intelligent Autonomous Systems (IAS)
- IEEE Potentials

INVITED LECTURES

- "Novel Algorithms for Estimating Camera Pose and Target Structure" – Lecture at Robotics Graduate Student Seminars, University of Illinois at Urbana Champaign, October 21, 2016
- "Novel Algorithms for Estimating Camera Pose and Target Structure" – Lecture at Computer Science and Engineering Graduate Student Seminars, Texas A&M University, September 12, 2016
- "Decentralized Formation Control and Obstacle Avoidance for Mobile Robots" – Plenary Talk, The 6th Robotics Workshop and Planning Movements, El Centro de Investigación en Matemáticas, Guanajuato, Mexico, April 29, 2016
- "Current Problems in Robot Vision and Control" – Lecture at Math Department Colloquium, Southern Methodist University, October 23, 2015

- “Real-Time Optimization with Applications in Visual Search” – Lecture at the Nonlinear and Dynamical Systems Symposium, University of Texas at Dallas, September 25, 2015
- “Real-Time Optimization with Applications in Visual Search” – Lecture at the Annual Meeting of the AFRL Mathematical Modeling and Optimization Institute, Eglin AFB July 29, 2014
- “Real-Time Optimization with Applications in Visual Search” – Lecture at Texas Systems Day, Texas A&M University, March 28, 2014
- “Real-Time Optimization for Visual Search” – Invited Lecture at Texas A&M Computer Science Graduate Seminar, October 22, 2012
- “Vision-based Control Beyond Position and Velocity Regulation” – Invited Lecture at the University of Texas at Arlington Electrical Engineering Graduate Seminar, February 3, 2012
- “Vision-Based Control Beyond Position and Velocity Regulation” Keynote Address, 22nd Annual International Symposium of the Pattern Recognition Association of South Africa, Vanderbijlpark, South Africa (November 2011)
- “Fusion of Vision and Inertial Measurement Units for Mobile Navigation” Dallas Chapter IEEE Signal Processing Society, (October 2010)
- “Balancing Mission Requirement for Networked Autonomous Aircrafts Performing Video Reconnaissance,” Department of Electrical Engineering, University of Texas at Dallas (April, 2009)
- “Balancing Mission Requirement for Networked Autonomous Aircrafts Performing Video Reconnaissance,” Department of Mechanical Engineering, Worcester Polytechnic Institute (March, 2009)
- “Balancing Mission Requirement for Networked Autonomous Aircrafts Performing Video Reconnaissance,” Department of Mechanical Engineering and Material Science, Duke University (January, 2009)
- “Underdetermined, Vision-Based Control of Mechanical Systems”, Sensors 2008: Theory, Algorithms, and Applications, University of Florida Research Engineering Education Facility (REEF) (April, 2008).
- “Underdetermined, Vision-Based Control of Mechanical Systems,” Department of Electrical and Computer Engineering, University of Denver (April, 2008)
- “Vision-Based Control of Mechanical Systems,” Department of Mechanical Engineering, Texas A&M University (February, 2008)
- “Simultaneous Stability of Image and Pose Error in Visual Servo Control”, Virginia Tech University (November, 2007)

PATENTS

1. W. Dixon, N. Gans, S. Gupta, *Passive Single Camera Imaging System for Determining Motor Vehicle Speed*, US Patent Number US 8401240 B2, Awarded March 19, 2013
2. W. Dixon, N. Gans, M. Kaiser, *Image-Based System and Methods for Vehicle Guidance and Navigation*, US Patent Number US 8,320,616 B2, Awarded November 27, 2012
3. N. Gans, W. Dixon Patent Cooperative Treaty No. PCT/US09/52803, *Systems and Methods for Maintaining Multiple Objects Within a Camera Field-of-View*, Publication number: US 2011/0128387 A1, filed June 2, 2011 – Received Notice of Allowance November 2016

PUBLICATIONS

Book Chapters

1. S. Mehta, G. Hu, N. Gans, and W. E. Dixon, “A Daisy-Chaining Visual Servoing Approach with Applications in Tracking, Localization, and Mapping,” in *Robot Localization and Map Building*, Chapter 20, pp. 383-408, Edited by A. Lazinica, In-Tech, March 2010.
2. N. Gans, G. Hu and W. E. Dixon, “Image-Based State Estimation” in Autonomous Robotics, Complexity and Nonlinearity, in *Encyclopedia of Complexity and Systems Science*, Springer, pp. 42-63, Vol. 1, 2009
3. G. Hu, N. Gans and W. E. Dixon, “Adaptive Visual Servo Control” in Autonomous Robotics, Complexity and Nonlinearity, in *Encyclopedia of Complexity and Systems Science*, Springer, p. 4751-4776, Vol. 5, 2009
4. P. I. Corke, S. A. Hutchinson and N. R. Gans, “Partitioned Image-Based Visual Servo Control: Some New Results,” in *Sensor Based Intelligent Robots, Springer Lecture Notes in Computer Science*, G. D. Hager, H. I. Christensen, H. Bunke, R. Klein Eds., Springer, 2002, pp. 122-140

Articles

5. T.R. Bennett, N. Gans, and J.R Jafari, “Data-Driven Synchronization for Internet-of-Things Systems,” *ACM Transactions on Embedded Computing Systems*, vol. 16, no. 3, p. 69.
6. J. Jin, Y. Kim and S. Wee, D. Lee and N. Gans, “A Stable Switched-System Approach to Collision-Free Wheeled Mobile Robot Navigation,” *Journal of Intelligent & Robotic Systems*, vol. 86, no. pp. 599-616, May.
7. E. A, Lee, M. Grohman, N. R. Gans, M. Tacca, & M. J. Brown, “The Roles of Implicit Understanding of Engineering Ethics in Student Teams’ Discussion,” *Science and Engineering Ethics*, pp. 1-20, Dec. 2016.
8. Y. Zhang, O. Makarenkov, N. Gans, “Extremum Seeking Control of a Nonholonomic System with Sensor Constraints,” *Automatica*, vol. 70, pp. 86-93, Aug. 2016
9. Y. Zhang, M. Rotea and N. Gans, “Simplex Guided Extremum Seeking Control with Convergence Detection to Improve Global Performance,” *IEEE Transaction on Control System Technology*, vol. 24, no. 4, pp. 1266-1278, July 2016.
10. J-P. Ramirez-Paredes, D. Lary and N. Gans, “Low-Altitude Terrestrial Spectroscopy from a Pushbroom Sensor,” *Journal of Field Robotics*, vol. 33, no. 6, Sept. 2016 pp. 837–852.
11. J. P. Ramirez-Paredes, E. A. Doucette, J. W. Curtis and N. R. Gans, "Optimal Placement for a Limited-Support Binary Sensor," in *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 439-446, Jan. 2016.
12. J. Jin, and N. Gans, “Parameter Identification for Industrial Robots with a Fast and Robust Trajectory Design Approach,” *Robotics and Computer Integrated Manufacturing*, vol 31, pp. 21-29, 2015,
13. W. MacKunis, N. Gans, A. Parikh, and W. E. Dixon, “Unified Tracking and Regulation Visual Servo Control for Wheeled Mobile Robots,” *Asian Journal of Control*, vol. 16, no.3, pp. 669- 678, 2014
14. Y. Zhang, J. Shen, and N. Gans, “Real-Time Optimization for Eye-in-Hand Visual Search,” *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 325 - 339, April 2014

15. D. Tick, A. C. Satici, J. Shen, and N. R. Gans, "Tracking Control of Mobile Robots Localized via Chained Fusion of Discrete and Continuous Epipolar Geometry, IMU and Odometry", *IEEE Transactions on Systems Man and Cybernetics Part B*, Volume 43, Number 4, 1237 – 1250, August 2013
16. N. R. Gans, G. Hu, J. Shen, Y. Zhang, and W. E. Dixon, "Adaptive Visual Servo Control to Simultaneously Stabilize Image and Pose Error," *Mechatronics*, Volume 22, Number 4, pp. 410–422, June 2012
17. N. R. Gans, G. Hu, K. Nagaragan, W. E. Dixon, "Keeping Multiple Moving Targets in the Field of View of a Mobile Camera," *IEEE Transactions on Robotics,* Volume 27, Number 4, pp. pp. 822 – 828, 2011
18. G. López-Nicolás, N. R. Gans, S. Bhattacharya, C. Sagüés, J.J. Guerrero and S. Hutchinson, "An Optimal Homography-Based Control Scheme for Mobile Robots with Nonholonomic and Field-of-View Constraints," *IEEE Transactions on Systems, Man and Cybernetics, Part B*, Volume 40, Number 4, pp. 1115 - 1127, 2010
19. M. K. Kaiser, N. Gans and W. Dixon, "Vision-Based Estimation and Control of an Aerial Vehicle through Chained Homography," *IEEE Transactions on Aerospace and Electronic Systems*, Volume 46, Number 3, pp. 1064-1077, 2010
20. G. Hu, N. Gans, and W. E. Dixon, "Quaternion-Based Visual Servo Control in the Presence of Camera Calibration Error," *International Journal of Robust and Nonlinear Control,* Volume 20, Number 5, pp. 489-503, 2010
21. G. Hu, N. R Gans, N. Fitz-coy, and W.E. Dixon, "Adaptive Homography-Based Visual Servo Tracking Control via a Quaternion Formulation," *IEEE Transactions on Control System Technology*, Volume 18, Number 1, pp. 128-135, 2010
22. N. Gans, W. Dixon, R. Lind and A. Kurdilla, "A Hardware in the Loop Simulation Platform for Vision-Based Control of Unmanned Air Vehicles," *IFAC Journal of Mechatronics* Volume 19, Number 7, 2009, pp. 1043-1056
23. G. Hu, W. MacKunis, N. Gans, W. E. Dixon, J. Chen, A. Behal, D. Dawson, "Homography-Based Visual Servo Control with Imperfect Camera Calibration," *IEEE Transactions on Automatic Control*, Volume 54, Number 6, pp. 1318-1325, 2009
24. N. R. Gans and S.A. Hutchinson, "Multi-Attribute Utility Analysis in the Choice of a Vision-Based Robot Controller," *International Journal of Optomechatronics*, Volume 2, Number 3, pp. 326-360, 2008
25. K. Dupree, N.R. Gans, W. MacKunis and W.E. Dixon, "Euclidean Calculation of Feature Points of a Rotating Satellite: A Daisy Chaining Approach," *AIAA Journal of Guidance, Controls, and Dynamics*, Volume 31, Number 4, pp. 954-961, 2008
26. N.R. Gans and S.A. Hutchinson, "Stable Visual Servoing through Hybrid Switched System Control," *IEEE Transactions on Robotics*, Volume 23, No. 3, pp. 530-540, 2007
27. N.R. Gans, P.I. Corke and S.A. Hutchinson, "Performance Tests for Visual Servo Control Systems, with Application to Partitioned Approaches to Visual Servo Control," *International Journal of Robotics Research*, volume 22, No. 10, pp 955-981, 2003

Invited Papers

28. Y. Zhang, M. Rotea, N.R. Gans, "Sensors Looking for Interesting Things: Extremum Seeking Control on Entropy Maps," *Proc. IEEE Conference on Decision and Control*, pp. 4985 - 4991, 2011

29. D.Q. Tick, J. Shen, Y. Zhang, N.R. Gans, “Chained Fusion of Discrete and Continuous Epipolar Geometry with Odometry for Long-Term Localization of Mobile Robots,” *Proc. IEEE International Symposium on Intelligent Control*, pp. 668 – 674, 2011
30. N. R. Gans, J. Shen, J. Shea, P. Barooah, W. Dixon “Balancing Mission Requirement for Networked Autonomous Quadrotor Performing Video Reconnaissance,” *Proc. SDPS Transformative Systems Conference*, 2010
31. N. R. Gans, G. Hu, W. E. Dixon, “Simultaneous Stability of Image and Pose Error in Visual Servo Control,” *Proc. IEEE International Symposium on Intelligent Control*, pp. 438-443, 2008
32. N. R. Gans, G. Hu, W. E. Dixon, “Keeping Objects in the Field of View: An Underdetermined Task Function Approach to Visual Servoing,” *Proc. IEEE International Symposium on Intelligent Control*, pp. 432-437, 2008
33. M. Kaiser, N.R. Gans and W.E. Dixon “Position and Orientation of an Aerial Vehicle through Chained, Vision-Based Pose Reconstruction,” *Proc. AIAA conference on Guidance, Navigation and Control*, 2006
34. S. Mehta, K. Kaiser, N.R. Gans, W. E. Dixon, “Homography-Based Coordinate Relationships for Unmanned Air Vehicle Regulation,” *Proc. AIAA conference on Guidance, Navigation and Control*, 2006
35. N.R. Gans and S.A. Hutchinson, “A Switching Approach to Visual Servo Control,” *Proc. IEEE International Symposium on Intelligent Control*, Workshop in Visual Servoing, Lausanne, 2002
36. N.R. Gans and S.A. Hutchinson, “Hybrid Visual Servo Control,” *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002, pp. 770-760

Proceedings of Technical Meetings

37. K. Fathian, D. Rachinskii, T. Summers, M. Spong and N. Gans, “Distributed Formation Control Under Arbitrarily Changing Topology”, *Proc. American Controls conference*, May 2017.
38. Y. Li, E. Doucette, J.W. Curtis and N. Gans, "Ground Target Tracking and Trajectory Prediction by UAV Using a Single Camera and 3D Road Geometry Recovery", ”, *Proc. American Controls conference*, May 2017
39. S. Kumar, and N. Gans, “Extremum Seeking Control for Multi-Objective Optimization Problems”, *Proc. IEEE Conf. Decision and Control*, pp. 1112-111, December 2016
40. K. Fathian, D. Rachinskii, T. Summers and N. Gans, “Distributed Control of Cyclic Formations with Local Relative Position Measurements”, *Proc. IEEE Conf. Decision and Control*, December 2016
41. O. Ogunmolu, X. Gu, S. Jiang, and N. Gans, “Vision-Based Control of a Soft Robot for Maskless Head and Neck Cancer Radiotherapy,” *Proc. IEEE International Conf. Automation Science and Engineering*, August 2016
42. J. P. Ramirez, E. Doucette, J. W. Curtis, N. Gans, “Optimal Placement for Limited-Support Binary Sensors,” *Proc. IEEE International conference on Robotics and Automation*, pp. 113–116, May, 2016
43. Y. Li and N. Gans, "Predictive RANSAC: An Effective Model Fitting and Tracking Approach with Application to Curve Fitting in Video", *Proc. American Controls Conference*, pp. 3620-3625, July, 2016.

44. K. Fathian, D. I. Rachinskii, M. W. Spong, N. R. Gans, "Globally Asymptotically Stable Distributed Control for Distance and Bearing Based Multi-Agent Formations *Proc. American Controls Conference*, pp. 4642 - 464, July, 2016.
45. M. J. Brown, E.-A. Lee, M. G. Grohman, N. Gans, and M. Tacca “It's Up to the Managers": The Deferral of Responsibility in Engineers' Discussion of Ethics,” *Proc. Consortium for Socially Relevant Philosophy of/in Science*, May 2016
46. E.-A. Lee, M. G. Grohman, N. Gans, M. Tacca, and M. J. Brown, “A Role of Implicit Understanding of Engineering Ethics in Student Teams’ Discussion ”, *Proc. National Association for Research in Science Teaching Annual Conference*, April 2016.
47. A. Iqbal, C. Busso, C. and N.R. Gans, “Adjacent Vehicle Collision Warning System using Image Sensor and Inertial Measurement Unit”, *Proc. International Conference on Multimodal Interaction*, pp. 291-298, nov., 2015.
48. T. R. Bennett, N. Gans, and R. Jafari. “Multi-sensor data-driven: synchronization using wearable sensors,” *In Proceedings of the 2015 ACM International Symposium on Wearable Computers*, pp. 113–116, September 7-11 2015
49. T. R. Bennett, N. Gans, and R. Jafari. “A data-driven synchronization technique for cyber-physical systems.” *In Workshop on the Swarm at the Edge of the Cloud*, April 13 2015.
50. O. Ogunmolu, X. Gu, S. Jiang, and N. Gans, “A Real-Time, Soft Robotic Patient Positioning System for Maskless Head-and-Neck Cancer Radiotherapy: An Initial Investigation,” *Proc. IEEE International Conf. Automation Science and Engineering,* August 2015
51. J. P. Ramirez, E. Doucette, J. W. Curtis, N. Gans, “Urban Target Search and Tracking Using a UAV and Unattended Ground Sensors,” *Proc. American Control Conference,* July 2015
52. J. Jin, Y. Kim and S. Wee, and N. Gans, “Consensus based attractive vector approach for formation control of nonholonomic mobile robots,” *Proc. IEEE International Conf. on Advanced Intelligent Mechatronics*, July 2015
53. E.-A. Lee, M. G. Grohman, N. Gans, M. Tacca, and M. J. Brown, “Situated and Expert-Guided Discussion of Engineering Ethics in Student Teams”, *Proc. American Society of Engineering Education Annual Conference, June 2015*
54. J. Jin, Y. Kim and S. Wee, and N. Gans, “Decentralized Cooperative Mean Approach to Collision Avoidance for Nonholonomic Mobile Robots,” *Proc. IEEE International Conf. on Robotics and Automation*, May 2015
55. J. Jin, Y.-G. Kim and S.-G. Wee, and N. Gans, “A Stable Switched-System Approach to Shared Robust Control and Obstacle Avoidance for Mobile Robots”, *Proc. of the ASME Dynamic Systems and Control Conferences*, 2014.
56. J. Jin, and N.R. Gans, “A Stable Switched-System Approach to Obstacle Avoidance for Mobile Robots in SE(2),” *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, 2014*
57. T. R. Bennett, N. Gans, and R. Jafari, “Motion Based Acceleration Correction for Improved Sensor Orientation Estimates,” *Proc. Body Sensors Networks Conference*, 2014
58. K. Fathian, J. Jin and N. Gans, “A New Approach for Solving the Five-Point Relative Pose Problem for Vision-Based estimation and Control,” *Proc. American Control Conference 2014*
59. P. Ramirez, E. Doucette, J.W. Curtis, and N. Gans, “Moving Target Acquisition Through State Uncertainty Minimization,” *Proc. American Control Conference 2014*

60. Y. Li, A.Iqbal, N.R. Gans, "Multiple lane boundary detection using a combination of low-level image features," *Proc. IEEE Conference on Intelligent Transportation Systems*, pp.1682-1687, 2014
61. S. Rodriguez, Y. Zhang, N. Gans, and N. Amato, "Optimizing aspects of pedestrian traffic in building designs," *Proc. IEEE/RSJ Symposium on Intelligent Robots and Systems, pp. 1327-1334*, 2014.
62. A. Satici, D. Tick, J. Shen, and N. Gans, "Path-Following Control For Mobile Robots Localized Via Sensor-Fused Visual Homography," *Proc. American Controls Conference. 2013*
63. T. Bennett, R. Jafari, and N. Gans, "An Extended Kalman Filter to Estimate Human Gait Parameters and Walking Distance," *Proc. American Controls Conference, 2013*
64. Y. Zhang and N. Gans, "Extremum Seeking Control of a Nonholonomic Mobile Robot with Limited Field of View," *Proc. American Controls Conference, 2013*
65. J. Shen, J. Jin and N. Gans "A Multi-View Camera-Projector System for Object Detection and Robot-Human Feedback," *Proc. of International Conference on Robotics and Automation*, 2013
66. J. Shen, J. Jin and N. Gans "A Trifocal Tensor Based Camera-Projector System for Robot-Human Interaction" *Proc. IEEE International Conference on Robotics and Biomimetics*, 2012
67. K. Fathian, F. Hassanipour, and N. R. Gans "Virtual Thermal Sensing And Control of Heat Distribution Using State Estimation," *Proc. ASME International Mechanical Engineering Congress & Exposition*, 2012
68. D. Tick, T. Rahman, C. Busso, N. Gans, "Robotic Terrain Classification Via Angular Velocity Based Hierarchical Classifier Selection," *International Conference on Robotics and Automation*, pp. 3594 – 3600, 2012
69. H. Poonawala, A. Satici, N. Gans, M. Spong, "Formation control of wheeled robots with vision-based position measurement," Proc. *American Control Conference*, pp. 3173-3178, 2012
70. Y. Zhang, N. Gans, "Simplex Guided Extremum Seeking Control," *Proc. American Control Conference*, pp. 3377 – 3382, 2012
71. Nagarajan, K., N.R. Gans and R. Jafari, "Modeling human gait using a Kalman Filter to measure walking distance", *Proc. Wireless Health Conference, 2011*
72. Y. Zhang, J. Shen, M. Rotea, N.R. Gans, "Robots Looking for Interesting Things: Extremum Seeking Control on Saliency Maps," *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1180 – 1186, 2011
73. P. Shankar, N. Raveendranathan, N.R. Gans, and R. Jafari. 2010. "Towards Power Optimized Kalman Filter for Gait Assessment Using Wearable Sensors" *Proc. Wireless Health Conference*, 137-144, 2010
74. J. Shen, D. Q. Tick, N.R. Gans, "Localization Through Fusion of Discrete and Continuous Epipolar Geometry with Wheel and IMU Odometry," *Proc. American Controls Conference*, 2010
75. K. Nagarajan, W. Yu, N.R. Gans, "Keeping Multiple Moving Objects in Field of View of a Mobile Robot," *Proc. IEEE International Workshop on Robotic and Sensors Environments*, pp.1-5, 2010

76. D. Q. Tick, J. Shen, and N.R. Gans, "Fusion of Discrete and Continuous Epipolar Geometry for Visual Odometry and Localization," *Proc. IEEE International Workshop on Robotic and Sensors Environments*, pp.1-6, 2010 (**Awarded Best Student Paper**)
77. N. R. Gans, J. Shen and J. W. Curtis, "Selection Of A UAV Orbit To Keep Multiple Targets In The Camera Field Of View," Proc. IEEE International Symposium on Intelligent Control (ISIC), pp.807-812, 2010
78. N. R. Gans, J.W. Curtis, J. Shea, P. Barooah and W. E. Dixon, "Balancing Mission Requirement for Networked Autonomous Rotorcrafts Performing Video Reconnaissance," *Proc. AIAA conference on Guidance, Navigation and Control*, 2009
79. N. R. Gans, G. Hu and W. E. Dixon, "Keeping Multiple Objects in the Field of View of a Single PTZ Camera," *Proc. American Controls Conference*, pp. 5259-5264, 2009
80. A. P. Dani, N. R. Gans and W. E. Dixon, "Position-Based Visual Servo Control of Leader-Follower Formation Using Image-Based Relative Pose and Relative Velocity Estimation," *Proc. of the American Control Conference*, pp. 5271-5276, 2009
81. N. Gans and J.W. Curtis, "A Performance Bound for Decentralized Moving Horizon Estimation," *Dynamics of Information Systems: Theory and Applications*, 2008
82. N. R. Gans, J. Shea, P. Barooah and W. E. Dixon, "Ensuring Network Connectivity of UAV's Performing Video Reconnaissance," *Proc. IEEE Military Communications Conference (MILCOM)*, 2008
83. A. P. Dani, S. Velat, C. Crane, N. R. Gans and W. E. Dixon, "Experimental Results for an Image-Based Pose and Velocity Estimation Method," *Proc. IEEE Conference on Control Applications*, pp. 1159-1164, 2008
84. R. Albertani, A. Chakravarthy and N. Gans, "Visual Flight Data and Frequency-Response Analysis Applied to Butterflies" (poster), *Proc. International Symposium on Adaptive Motion of Animals and Machines*, 2008
85. N. R. Gans, A. Dani, W. E. Dixon, *"*Visual Servoing to an Arbitrary Pose with Respect to an Object Given a Single Known Length,*" Proc. of the American Control Conference*, pp. 1261-1267, 2008
86. G. Hu, W. Mackunis, N. Gans, and W. E. Dixon, " Homography-Based Visual Servo Control via An Uncalibrated Camera," *Proc. American Controls Conference*, 2008, pp. 4791-4796
87. G. Hu, N.R. Gans, W.E. Dixon, "Quaternion-Based Visual Servo Control in the Presence of Camera Calibration Error," *IEEE Multi-conference on Systems and Control*, 2007, pp. 1492-1497
88. G. Hu, N.R. Gans, S. Mehta, W.E. Dixon, "Daisy Chaining Based Visual Servo Control Part I: Adaptive Quaternion-Based Tracking Control," *Proc. IEEE Multi-conference on Systems and Control*, 2007 pp. 1474-1479
89. G. Hu, S. Mehta, N.R. Gans, W.E. Dixon, "Daisy Chaining Based Visual Servo Control Part II: Extensions, Applications and Open Problems," *Proc. IEEE Multi-conference on Systems and Control*, 2007, pp. 729-734
90. W. Mackunis, N.R. Gans, M. Kaiser, W. E. Dixon, "Unified Tracking and Regulation Visual Servo Control for Wheeled Mobile Robots," *Proc. IEEE Multi-conference on Systems and Control*, 2007, pp. 88-93
91. K. Dupree, N.R. Gans, W. Mackunis, W. E. Dixon, "Euclidean Calculation of Feature Points of a Rotating Satellite: A Daisy Chaining Approach," *Proc. American Controls Conference*, 2007, pp. 5934-5939

92. K. Kaiser, N.R. Gans, W.E. Dixon, “Localization and Control of an Aerial Vehicle through Chained, Vision-Based Pose Reconstruction,” *Proc. American Controls Conference*, 2007, pp. 3874-3879
93. N.R. Gans, S.A. Hutchinson, “A Stable Vision-Based Control Scheme for Nonholonomic Vehicles,” *Proc. IEEE International Conf. on Robotics and Automation*, 2007, pp. 2196 - 2201
94. S.S. Mehta, G. Hu, N.R. Gans, W.E. Dixon, “Vision-Based Collaborative Tracking Control of Unmanned Ground Vehicles,” *Proc. IEEE Conference on Decision and Control*, 2006, pp. 3867-3872.
95. N.R. Gans and S.A Hutchinson, “Visual Servo Velocity and Pose Control of a Wheeled Inverted Pendulum through Partial-Feedback Linearization,” *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 3823-3828
96. N.R. Gans and S.A. Hutchinson, “Multi-Attribute Utility Analysis in the Choice of a Vision-Based Robot Controller,” *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004, pp. 355-362
97. B. Chambers, N.R. Gans, J. Durand and S.A. Hutchinson, “Dynamic feature point detection for visual servoing using multiresolution critical-point filters,” *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003, pp. 504–509
98. N.R. Gans and S.A. Hutchinson, “An Asymptotically Stable Switched System Visual Servo Controller for Eye in Hand Robotics,” *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003, pp. 735 - 742
99. N.R. Gans and S.A. Hutchinson, “An Experimental Study of Hybrid Switched System Approaches to Visual Servoing,” *Proc. IEEE International Conf. on Robotics and Automation*, 2003, pp. 3061 - 3068
100. N.R. Gans, P.I. Corke and S.A. Hutchinson, “Performance Tests of Partitioned Approaches to Visual Servo Control,” *Proc. IEEE International Conf. on Robotics and Automation*, 2002, pp. 1616-1623
101. N.R. Gans, P.I. Corke and S.A. Hutchinson, “Comparison of Robustness and Performance of Partitioned Image Based Visual Servo Systems,” *Proc. of Australian Conference on Robotics and Automation*, Sydney, 2001