

การหาทิศทางของอุปกรณ์สื่อสารเคลื่อนที่ที่ใช้ HTML5 สำหรับเทคโนโลยีเสมือนจริง

Direction Finding of Mobile Using HTML5 for Augmented Reality Technology

ปิยะพงษ์ แดงขำ

Piyapong Dangkhom

สาขาวิชาวิศวกรรมสารสนเทศและการสื่อสาร คณะเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยราชภัฏเทพสตรี

E-mail: piyapongdk@gmail.com

บทคัดย่อ

บทความนี้นำเสนอการหาทิศทางของอุปกรณ์สื่อสารเคลื่อนที่ด้วย HTML5 – Geolocation API ซึ่งการหาทิศทางเป็นขั้นตอนหนึ่งในการพัฒนาเทคโนโลยีเสมือนจริง โดยได้นำ coords.heading ที่ใช้ในระบบ GPS มาใช้งาน ผลการทดลองพบว่าการระบุทิศทางมีความผิดพลาดเฉลี่ย 5.8 องศา เมื่อเทียบกับการระบุทิศทางจากเข็มทิศ และจำนวนครั้งเฉลี่ยที่ใช้ในการคำนวณมีค่าเป็น 5.7 ครั้งต่อการระบุทิศทาง

คำสำคัญ: เทคโนโลยีเสมือนจริง, HTML5, Geolocation API

Abstract

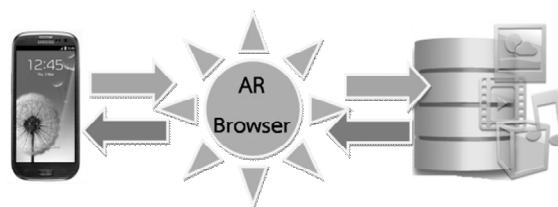
This paper presents direction finding of mobile using HTML5 – Geolocation API. Direction finding is the one of the augmented reality's process. The attribute of Geolocation API, coords.heading, is used in this work. The result shows that the error of finding direction is 5.8 degrees compared with compass and the times of calculation is 5.7.

Keywords: Augmented Reality, HTML5, Geolocation API

1. บทนำ

เทคโนโลยีเสมือนจริง (Augmented Reality: AR) เป็นเทคโนโลยีความจริงเสมือนประเภทหนึ่งที่มีการนำมาผนวกกับเทคโนโลยีภาพ โดยอาศัยการเพิ่มภาพเสมือนของโมเดลสามมิติ ซึ่งสร้างจากคอมพิวเตอร์ลงไปบนภาพถ่ายจากกล้องวิดีโอ หรือกล้องในอุปกรณ์สื่อสารเคลื่อนที่แบบเฟรมต่อเฟรม การนำเทคโนโลยี AR มาใช้บนอุปกรณ์สื่อสารเคลื่อนที่ได้รับความนิยมสนใจมาโดยตลอด เนื่องจากอุปกรณ์สื่อสารเคลื่อนที่ที่มีผู้ใช้งานจำนวนมาก และสามารถพกพาได้อย่างสะดวก การใช้งานจะเป็นลักษณะใช้กล้องวิดีโอถ่ายภาพในระดับสายตา แล้วข้อมูลของสถานที่ที่จะปรากฏให้เห็นบนหน้าจอ โดยผู้พัฒนาโปรแกรม Layar ซึ่งนับเป็น AR browser รายแรกของโลก[1] ได้นำเสนอเทคโนโลยี AR ในลักษณะที่กล่าวมานี้เชื่อมโยงกับข้อมูลของสถานที่ที่แสดง เช่น เว็บไซต์ หรือหมายเลขโทรศัพท์ที่เกี่ยวข้องซ้อนทับบนสถานที่นั้นๆ นอกจากนี้ยังมีผู้พัฒนาโปรแกรม AR Browser เพิ่ม

อีก เช่น junaio Wikitude Aurasma ซึ่งแต่ละผู้ผลิตจะพัฒนาโปรแกรมให้ครอบคลุมระบบปฏิบัติการบนอุปกรณ์สื่อสารเคลื่อนที่ทุกรุ่น เพื่อความสะดวกของผู้ใช้งาน หลักการทำงานของโปรแกรม AR Browser จะแสดงได้ดังภาพที่ 1 โดยขั้นตอนจะสัมพันธ์กันระหว่าง 3 ส่วนหลัก คือ อุปกรณ์สื่อสารเคลื่อนที่ โปรแกรม AR Browser และเครื่องแม่ข่าย



ภาพที่ 1 หลักการทำงานของโปรแกรม AR Browser

การทำงานของโปรแกรม AR Browser จะเริ่มต้นจากอุปกรณ์สื่อสารเคลื่อนที่เรียกใช้งานโปรแกรม AR Browser แล้วโปรแกรมจะทำการดึงข้อมูลที่ถูกจัดเก็บไว้ในเครื่องแม่ข่ายมาประมวลผล เช่น ชื่อสถานที่ หรือตำแหน่งที่ตั้ง โดยจะใช้หลักการประมวลผลภาพมาจัดการวางผลลัพธ์ที่ได้ซ้อนทับกับภาพจริง จากนั้นจึงทำการแสดงผลผ่านทางจอภาพของอุปกรณ์สื่อสารเคลื่อนที่

การพัฒนาโปรแกรมบนอุปกรณ์สื่อสารเคลื่อนที่ที่จะแตกต่างกันไปตามระบบปฏิบัติการที่รองรับ เช่น จะใช้ภาษา Java สำหรับการพัฒนาบนระบบปฏิบัติการ Android หรือใช้ภาษา Objective-C สำหรับการพัฒนาบนระบบปฏิบัติการ iOS แต่ในปัจจุบันมีการนำเทคโนโลยี HTML5 มาพัฒนาโปรแกรมร่วมด้วย โดยเทคโนโลยี HTML5 ถูกสร้างขึ้นเพื่อพัฒนาเว็บไซต์ที่ไม่ต้องการส่วนขยาย (Plug-in) ทำให้ผู้ใช้งานไม่จำเป็นต้องติดตั้งโปรแกรมอื่นๆ เพิ่ม เมื่อนำข้อดีของเทคโนโลยี HTML5 มาวิเคราะห์แล้ว จึงมีแนวความคิดที่จะนำมาพัฒนาโปรแกรมเสมือนจริง ซึ่งจะทำให้การแสดงผลจะไม่ถูกจำกัดอยู่ที่โปรแกรม AR Browser เพียงแค่อุปกรณ์สื่อสารเคลื่อนที่ที่สามารถเปิดเว็บไซต์ผ่านโปรแกรมเว็บเบราว์เซอร์ (Web Browser) เช่น Chrome Firefox หรือ Opera ได้ ก็จะสามารถใช้งานโปรแกรมเสมือนจริงได้ด้วย

2. การใช้ HTML5 ในเทคโนโลยีเสมือนจริง

การพัฒนาเว็บไซต์ได้มีการสร้างมาตรฐานภาษาขึ้นเรียกว่า HTML5 ซึ่งเป็นภาษาที่ไม่จำเป็นต้องพึ่งพาเทคโนโลยีอื่นมาร่วมทำงาน เช่น Flash Player เป็นต้น มาตรฐานของ HTML5 มีการกำหนดหน้าที่อย่างน้อย 2 ส่วน [2] คือ ส่วนการแสดงผลข้อมูล ซึ่งจะเหมือนกับ HTML รุ่นก่อนหน้า และอีกส่วนคือ ส่วนโครงสร้างข้อมูล ซึ่งจะเหมือนกับโครงสร้าง XML ในขณะที่โปรแกรมเว็บเบราว์เซอร์ของผู้ผลิตต่างๆ ก็มุ่งพัฒนาให้รองรับกับมาตรฐาน HTML5 นี้ด้วย ซึ่งตามมาตรฐานของ HTML5 จะทำการแบ่งข้อมูลแต่ละส่วนด้วยอีเลเมนต์ (element) ซึ่งเป็นข้อดีของ HTML5 ที่ช่วยให้การพัฒนาเว็บไซต์เป็นไปด้วยความมีระเบียบและง่ายต่อการแก้ไข นอกจากนี้ในแต่ละอีเลเมนต์ ยังมีคุณสมบัติ (attribute) เป็นตัวกำหนดการใช้งานให้หลากหลายมากขึ้นอีก ในปัจจุบันได้มีการนำเทคโนโลยี HTML5 มาพัฒนาโปรแกรมบนอุปกรณ์สื่อสารเคลื่อนที่ในรูปแบบ Cross-platform [3] ซึ่งเป็นการพัฒนาโดยไม่ต้องใช้ภาษาดั้งเดิม (Native) ของระบบปฏิบัติการนั้น ๆ แต่การใช้งาน HTML5 จะมีข้อจำกัดในการเข้าถึงการใช้งานอุปกรณ์บางตัว เช่น ระบบเซ็นเซอร์ กล้อง ระบบไฟล์เก็บข้อมูล เข็มทิศ เป็นต้น อีกทั้งยังขึ้นอยู่กับโปรแกรมเว็บเบราว์เซอร์ที่ใช้งานอีกด้วย การพัฒนาเทคโนโลยีเสมือนจริงโดยใช้ HTML5 จะต้องตรวจสอบตำแหน่งของผู้ใช้งานให้ได้ก่อนแล้วจึงทำการซ้อนทับด้วยภาพที่ต้องการลงไป เนื่องจาก HTML5 เป็นเทคโนโลยีเว็บจึงมีข้อจำกัดหลายอย่าง ประการแรกคือการเข้าถึงได้ของโปรแกรมเว็บเบราว์เซอร์แต่ละผู้ผลิต และประการที่สองคือการใช้งานระบบตรวจจับของอุปกรณ์ที่ไม่สามารถใช้ได้อย่างสมบูรณ์

2.1. การตรวจจับทิศทางจากเซ็นเซอร์ด้วย HTML5

เทคโนโลยีเสมือนจริงจำเป็นต้องรู้ตำแหน่งที่อยู่ของผู้ใช้งานให้ถูกต้องแล้วจึงนำไปผสานกับภาพที่สร้างขึ้นมา โปรแกรม AR Browser ที่พัฒนาขึ้นจะยืนยันตำแหน่งผ่านระบบเซ็นเซอร์ของอุปกรณ์ เช่น ระบบตรวจจับ GPS ระบบตรวจจับ Orientation หรือ ระบบตรวจจับ Magnetic แต่ปัญหาหลักที่นำ HTML5 มาพัฒนาเทคโนโลยีเสมือนจริงคือ การตรวจจับทิศทางเคลื่อนที่ของผู้ใช้งาน ซึ่งจะทำให้ยากกว่าการพัฒนาด้วยภาษาดั้งเดิมของแต่ละระบบปฏิบัติการ

จาก [4] ได้นำเสนอการตรวจจับทิศทางของอุปกรณ์สื่อสารเคลื่อนที่ด้วย HTML5 โดยเรียกใช้งานเซ็นเซอร์ 2 ชนิดคือ Orientation และ Accelerometer ซึ่งสามารถระบุทิศทางของผู้ใช้งานอ้างอิงกับทิศเหนือได้ แต่มีข้อจำกัดคือโปรแกรมเว็บเบราว์เซอร์ที่รองรับการใช้งานแสดงผลที่แตกต่างกันออกไป เช่น Chrome จะอ้างอิงกับทิศเหนือที่ 0 องศาแล้วหมุนวนทิศตามเข็มนาฬิกา ในขณะที่ Firefox จะอ้างอิงแบบทวนเข็มนาฬิกา ทำให้เกิดความสับสนของผู้ใช้งานโปรแกรมเว็บเบราว์เซอร์ที่ต่างกัน ผู้พัฒนาโปรแกรมเทคโนโลยีเสมือนจริงจึงต้องคำนึงถึงข้อจำกัด

ด้านนี้ ในขณะที่โทรศัพท์บางรุ่นไม่มีระบบตรวจจับ Orientation ก็จะสามารถหาทิศทางได้อย่างถูกต้องอีกด้วย

2.2. การใช้ HTML5 – Geolocation

มาตรฐาน HTML5 ได้เตรียม API ที่ใช้อ้างอิงตำแหน่งจากระบบ GPS คือ Geolocation ซึ่งมักจะถูกใช้งานร่วมกับการลงพิกัดบนแผนที่ เช่น Google Maps โดยการใช้งาน Geolocation API สามารถใช้งานได้ทุกโปรแกรมเว็บเบราว์เซอร์ และการแสดงผลจะได้ผลลัพธ์ที่เหมือนกันอีกด้วย เพียงแต่ค่าความละเอียดที่ได้จากระบบ GPS จะแตกต่างกันออกไปตามเซ็นเซอร์ของอุปกรณ์สื่อสารเคลื่อนที่

จากคุณสมบัติของ Geolocation API ที่ทุกโปรแกรมเว็บเบราว์เซอร์รองรับได้ จึงมีแนวความคิดที่จะนำมาพัฒนาหาทิศทางเคลื่อนที่ของอุปกรณ์สื่อสาร

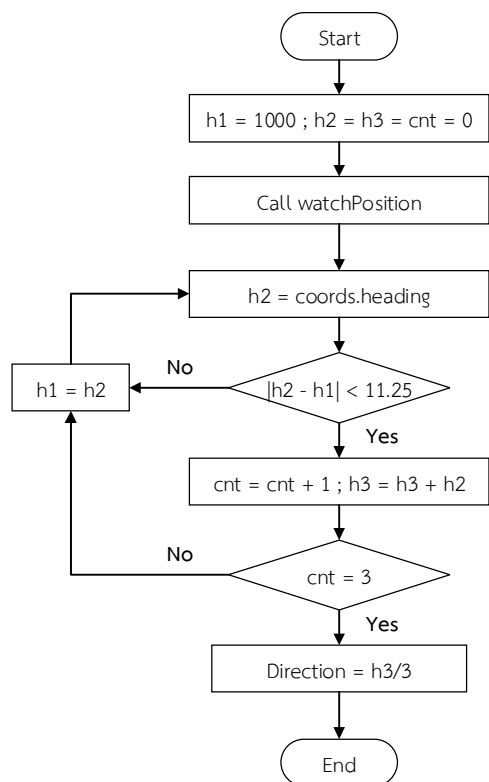
3. การหาทิศทางของอุปกรณ์สื่อสารเคลื่อนที่

Geolocation API มีอินเตอร์เฟซที่น่าสนใจคือ Geolocation และ Position โดยอินเตอร์เฟซ Geolocation มีหน้าที่เป็นตัวหลักในการรับข้อมูลตำแหน่งของอุปกรณ์ ซึ่งมีเมธอด watchPosition สำหรับแจ้งข้อมูลเมื่อตำแหน่งทาง GPS มีการเปลี่ยนแปลง ในขณะที่อินเตอร์เฟซ Position มีคุณสมบัติให้ใช้งานที่น่าสนใจดังแสดงในตารางที่ 1 [5]

ตารางที่ 1 คุณสมบัติของอินเตอร์เฟซ Position

ชื่อคุณสมบัติ	ประเภทหน่วย	ความหมาย
coords.latitude	double:degree	ตำแหน่งละติจูด
coords.longitude	double:degree	ตำแหน่งลองจิจูด
coords.accuracy	double:meter	ค่าความละเอียด
coords.heading	double:degree CW	ทิศทางเคลื่อนที่อ้างอิงทิศเหนือ
coords.speed	double:meter/second	ความเร็วในการเคลื่อนที่

จากตารางที่ 1 คุณสมบัติของอินเตอร์เฟซ Position ที่สามารถนำมาใช้ในการหาทิศทางเคลื่อนที่ได้คือ coords.heading โดยค่าที่ได้จะเป็นการอ้างอิงจากทิศเหนือ (0 องศา) และหมุนรอบตามเข็มนาฬิกา ทำให้การอ่านค่าที่ได้ในทิศตะวันออก ทิศใต้ และทิศตะวันตกเป็น 90 180 และ 270 องศาตามลำดับ แต่ข้อจำกัดในการใช้งาน coords.heading คือจะต้องมีการเปลี่ยนแปลงตำแหน่งทาง GPS เพื่อให้ทราบถึงทิศทางเคลื่อนที่ โดยมีขั้นตอนการหาทิศทางดังภาพที่ 2



ภาพที่ 2 ขั้นตอนการหาทิศทางด้วย Geolocation API

ขั้นตอนการหาทิศทางจะกำหนดตัวแปรที่เก็บค่าข้อมูลของ coords.heading เป็น $h1$ $h2$ และ $h3$ เมื่อ $h1$ และ $h2$ เป็นค่าที่ได้จากการเปลี่ยนแปลงตำแหน่งในแต่ละครั้ง และ $h3$ เป็นตัวแปรที่เก็บสะสมค่า coords.heading ที่ผ่านการตรวจสอบแล้ว ค่าผลต่างของค่า coords.heading ถูกตั้งเงื่อนไขไว้ที่น้อยกว่า 11.25 องศา พิจารณาจากการแบ่งขนาดของมุมระหว่างทิศทางที่ติดกันออกเป็นสี่ส่วน เพื่อให้มีช่องว่าง (gap) ระหว่างทิศทางที่ติดกัน และป้องกันไม่ให้เกิดการสับสนในการตัดสินใจทิศทาง โดยจะตรวจสอบค่าที่ถูกต้องทั้งหมด 3 ครั้ง แล้วจึงทำการหาค่าเฉลี่ยเพื่อให้ได้ทิศทางที่เคลื่อนที่

4. ผลการทดลอง

ในขั้นตอนทดลองหาทิศทางที่เคลื่อนที่ด้วยการใช้ HTML5 - Geolocation จะนำเข็มทิศเป็นอุปกรณ์อ้างอิงทิศทางที่ถูกต้อง จากนั้นจึงเคลื่อนที่ไปตามทิศทางดังกล่าวเพื่อตรวจสอบความผิดพลาดในการใช้งาน การบันทึกค่าการหาทิศทางแสดงไว้ในภาพที่ 3

Direction: N E S W NE NW SE SW

Test

#	h1	h2	h2-h1	h3
1	1000	null	null	0
2	1000	null	null	0
3	1000	174.2	825.8	0
4	174.2	174.4	0.2	174.4
5	174.4	176.3	1.9	350.7
6	176.3	175.5	0.8	526.2

Direction: $526.2/3 = 175.4$ degree
Error: $180 - 175.4 = 4.6$ degree

ภาพที่ 3 ผลการทดลองหาทิศทางในทิศได้

จากภาพที่ 3 แสดงผลการทดลองหาทิศทางที่เคลื่อนที่ไปในทิศได้ โดยอ้างอิงเส้นทางด้วยเข็มทิศ และเงื่อนไขการคำนวณเป็นไปตามที่ได้เสนอไว้ในหัวข้อที่ 3 ซึ่งจะพบว่าบางครั้งไม่สามารถหาค่า coords.heading ได้ ($h2$ มีค่าเป็น null) จึงทำให้จำนวนครั้งในการค้นหาทิศทางเพิ่มมากขึ้น ค่า $h3$ จะมีการเปลี่ยนแปลงเมื่อผลต่างของ $h2$ และ $h1$ มีค่าน้อยกว่า 11.25 โดยจะเก็บค่าเป็นผลบวกสะสมของค่า $h2$ ที่ได้รับเนื่องจากการเปลี่ยนตำแหน่งในแต่ละครั้ง จากนั้นจึงนำมาหาค่าเฉลี่ย แล้วทำการเปรียบเทียบกับทิศทางที่อ้างอิงจากเข็มทิศ ในการทดลองจะทำการเคลื่อนที่ไปในทิศทางทั้ง 8 ทิศ จำนวนทิศทางละ 10 ครั้ง แล้วจึงหาจำนวนครั้งที่ใช้ในการระบุทิศทาง และค่าความผิดพลาดเมื่ออ้างอิงด้วยเข็มทิศ ผลการทดลองแสดงได้ดังตารางที่ 2

ตารางที่ 2 ผลการทดลองหาทิศทางที่เคลื่อนที่

ทิศทาง	จำนวนครั้ง (ครั้ง)	ค่าความผิดพลาด (องศา)
เหนือ	5.5	5.4
ตะวันออกเฉียงเหนือ	6.0	5.2
ตะวันออก	6.0	6.2
ตะวันออกเฉียงใต้	5.4	7.2
ใต้	5.7	4.8
ตะวันตกเฉียงใต้	5.6	6.3
ตะวันตก	5.7	5.4
ตะวันตกเฉียงเหนือ	5.6	5.5

จากตารางที่ 2 จะพบว่าจำนวนครั้งในการคำนวณโดยเฉลี่ยทั้ง 8 ทิศทางเป็น 5.7 ครั้ง และมีค่าความผิดพลาดเฉลี่ยอยู่ที่ 5.8 องศา เมื่ออ้างอิงกับทิศทางที่ได้จากเข็มทิศ

5. สรุปผลการวิจัย

บทความนี้ได้นำเสนอการใช้ HTML5 พัฒนาโปรแกรมเทคโนโลยีเสมือนจริง ซึ่งมีข้อดีในเรื่องของการใช้งานโดยไม่ต้องติดตั้งโปรแกรมอื่นๆ เพิ่ม สามารถใช้งานผ่านโปรแกรมเว็บ



บราวเซอร์ได้ โดยได้นำเสนอในส่วนของการหาทิศทาง การเคลื่อนที่ของผู้ใช้งานผ่าน Geolocation API ด้วยการเรียกใช้คุณสมบัติ coords.heading ผลการทดลองพบว่าสามารถนำคุณสมบัติดังกล่าวมาใช้ในการบอกทิศทางเคลื่อนที่ได้ โดยมี ความผิดพลาดเฉลี่ย 5.75 องศา และต้องใช้จำนวนครั้งในการคำนวณเฉลี่ยที่ 5.88 ครั้ง ดังนั้น HTML5 – Geolocation จึงสามารถนำมาใช้งานในเทคโนโลยีเสมือนจริงแทนการเรียกใช้ HTML5 ติดต่อบริบบ์เซิร์ฟเวอร์โดยตรง ซึ่งอุปกรณ์เคลื่อนที่บางรุ่น ไม่มีการติดตั้งไว้ในตัวเครื่อง

6. เอกสารอ้างอิง

- [1] พนิดา ต้นศิริ, “โลกเสมือนผสมผสานโลกจริง (Augmented Reality),” วารสารนักบริหารมหาวิทยาลัยกรุงเทพ,ปีที่ 32 ฉบับที่ 2, 2555.
- [2] ศุภชัย สมพานิช, “Basic HTML5 with CSS3 & jQuery,” ไอทีชีวะ, นนทบุรี, 2556.
- [3] กอบเกียรติ สระอุบล, “พัฒนา Cross-Platform Mobile App สำหรับ iOS Android,” ซีอีโอเคชั่น, กรุงเทพฯ, 2557.
- [4] Ruadhán, “HTML5 for the Mobile Web: Device Orientation Events,” [online], เข้าถึงจาก <http://mobiforge.com/design-development/html5-mobile-web-device-orientation-events,1> กุมภาพันธ์ 2558.
- [5] Andrei Popescu, “Geolocation API Specification,” [online], เข้าถึงได้จาก <http://dev.w3.org/geo/api/spec-source.html,12> กุมภาพันธ์ 2558.