บทความวิจัย

Computer Science Department
Faculty of Informatics, Mahasarakham University

ระบบตรวจจับภาวะหลับใน

Drowsiness Detection System

กรวิชญ์ อุทกโยธะ, ชิตพล จันทร์อาภาท, นัฐธริยา เหล่าประชา
สาขาวิทยาการคอมพิวเตอร์ คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหารคาม
63011212022@msu.ac.th, 63011212079@msu.ac.th, natthariya@gmail.com

บทคัดย่อ

การพัฒนาระบบตรวจจับภาวะหลับในที่ตรวจจับภาวะหลับในขณะขับรถด้วยการเรียนรู้เชิงลึก มีจุดมุ่งหมายเพื่อที่จะลดการเกิดอุบัติเหตุที่และลดการสูญเสียที่เกิดจากภาวะหลับในในขณะขับรถที่เป็นสาเหตุหลักบนท้องถนน

วิธีที่ใช้ทดลองใช้วิดีโอจำนวน 20 ชุด ในการทดลองแต่ละวิดีโอมีความยาว 1 – 2 นาที และในวิดีโอจะมีผู้ขับขี่ที่เกิดภาวะหลับในและไม่เกิดเข้ามาทดลองโดยเราจะใช้การตรวจจับดวงตาเพื่อตรวจจับภาวะหลับใน เมื่อผู้ขับขี่มีอาการหลับในดวงตาของผู้ขับขี่จะปิด เมื่อปิดนานเกิน 3 วินาที โปรแกรมจะแจ้งเตือนผู้ขับขี่ว่าหลับใน จากผลการทดลองจากวิดีโอทั้งหมด 20 วิดีโอ มีความถูกต้อง 52 %

1.บทนำ

ในปัจจุบัน การใช้รถใช้ถนนนั้นมีการเพิ่มเยอะขึ้น เนื่องจากการเดินทางด้วยรถยนต์และรถจักรยานยนต์ นั้นมีความสะดวกสบายและมีความรวดเร็ว เนื่องจากมีประชากรเพิ่มมากขึ้นแต่ต้องการซื้อรถเพื่อเอาไว้เดินทาง แต่ยังไงก็มีข้อเสียอยู่บ้าง ต้องตรวจเช็คสภาพรถก่อนการเดินทาง รวมทั้งการล้างทำความสะอาดรถ ซึ่งในช่วงหนึ่งของทุก ๆ ปี ก็จำเป็นต้องนำรถไปเข้าศูนย์ เพื่อเช็คสภาพด้วย และถ้ามีส่วนไหนเสียหายก็ต้องเสียค่าซ่อมบำรุง แต่ก็ปฏิเสธไม่ได้ว่ามีอุบัติเหตุบนท้องถนนที่เกิดจากรถยนต์และรถจักรยานยนต์ก็เพิ่มมากขึ้นตามไปด้วย อีกนึงสาเหตุสำคัญที่ทำให้เกิดอุบัติเหตุบนท้องถนนบ่อยครั้ง ก็คือภาวะหลับใน หรือ อาการหลับใน ภาวะหลับในหรือการหลับระยะสั้น ๆ (Microsleep) เป็นปรากฏการณ์การสับสนระหว่างการหลับและการตื่น โดยการหลับเข้ามาแทรกการตื่นอย่างเฉียบพลันโดยไม่รู้ตัวในช่วงเวลาสั้น ๆ ประมาณ 1 – 2 วินาที ภาวะหลับในเกิดขึ้นได้จากหลายสาเหตุ เช่น การอดนอน นอนไม่เป็นเวลา จะเพิ่มความเสี่ยงในการเกิดภาวะหลับในเพราะสมองส่วนธาลามัสอาจหยุดทำงานสั้น ๆ ชั่วคราว ส่งผลให้เกิดความง่วงกะทันหัน (Sleep Attack) งีบหลับไม่รู้ตัว ไม่ตอบสนองต่อการรับรู้จนเกิดภาวะหลับในได้ ทำให้เกิดอุบัติเหตุได้

ผู้พัฒนาได้เล็งเห็นการบาดเจ็บและเสียชีวิตจากเกิดอุบัติเหตุ เพราะภาวะหลับในจึงได้พัฒนาระบบตรวจสอบอาการง่วงนอนซึ่งจะมีเสียงแจ้งเตือนผู้ขับขี่เป็นระยะ เพื่อป้องกัน

Computer Science Department
Faculty of Informatics, Mahasarakham University

ผู้ขับขี่จากอุบัติเหตุ โดยติดตั้งกล้องไว้ในรถเพื่อบันทึกภาพวิดีโอของผู้ขับขี่ การตรวจจับใบหน้าใช้อัลกอริทึมวิเคราะห์สีผิว และการตรวจหาดวงตาใช้หลักการหาพื้นที่ที่พบพิกเซลสีขาวบนใบหน้า ระบบจะตรวจสอบสถานะดวงตาเพื่อหาว่าตาเปิดหรือปิด ถ้าพบว่าตาปิดนานกว่าปกติค่าความง่วงนอนจะเพิ่มขึ้น ในทางตรงกันข้ามค่าความง่วงนอนจะลดลง ถ้าค่าความง่วงนอนมีค่ามากกว่าหรือเท่ากับค่าเกณฑ์ที่กำหนด ระบบยังช่วยตรวจสอบและวิเคราะห์พฤติกรรมการขับขี่ของผู้ขับขี่ เช่น ความเร็วสูงเฉียบพลันในเวลาที่สั้น การเปลี่ยนแปลงช่องทางที่ผิดปกติด้วยความเร็วที่ผิดปกติ ระบบจะส่งสัญญานเตือนอย่างต่อเนื่อง ทำให้ผู้ขับขี่ตื่นตัวและช่วยลดการเกิดอุบัติเหตุได้

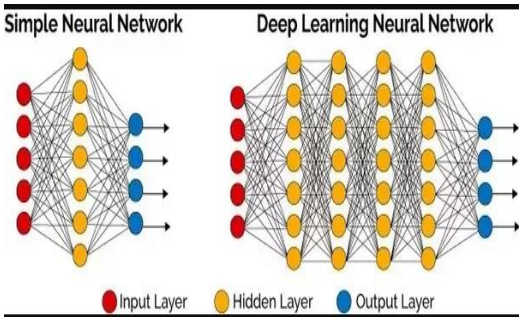
2.ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ปัญญาประดิษฐ์

AI ย่อมาจาก Artificial Intelligence โดยภาษาไทยใช้คำว่า ปัญญาประดิษฐ์ หมายถึง ระบบประมวลผลของคอมพิวเตอร์ หุ่นยนต์ เครื่องจักร หรืออุปกรณ์อิเล็กทรอนิกส์ต่าง ๆ ที่มีการวิเคราะห์เชิงลึกคล้ายความฉลาดของมนุษย์ และสามารถก่อให้เกิดผลลัพธ์ที่เป็นการกระทำได้ กระบวนการเรียนรู้ของ AI ไม่ต่างจากการเรียนรู้ของมนุษย์ เป็นกระบวนการจดจำ ทำความเข้าใจ ตอบสนองต่อภาษา ตัดสินใจ และแก้ไขปัญหา โดยอาศัยข้อมูลจำนวนมากที่มีลักษณะซ้ำ ๆ เหมือนกันทั้งนี้การใช้ AI ที่ถูกต้องเหมาะสม และก่อให้เกิดประโยชน์สูงสุดจำเป็นต้องมีการวิเคราะห์และเลือกสรรให้สอดคล้องกับจุดประสงค์ของการใช้ เพราะฉะนั้นจึงสามารถกล่าวได้ว่า AI ถือกำเนิดขึ้นเมื่อเครื่องจักรมีความสามารถที่จะเรียนรู้นั่นเอง ซึ่ง AI ก็ถูกแบ่งออกเป็นหลายระดับตามความสามารถหรือความฉลาด โดยจะวัดจากความสามารถในการ ให้เหตุผล การพูด และทัศนคติของ AI ตัวนั้น ๆ เมื่อเปรียบเทียบกับมนุษย์อย่างเรา ๆ AI จะถูกแบบออกเป็น 2 องค์ประกอบ ได้แก่ 1. Machine Learning คือ ส่วนที่เปรียบเสมือนสมองของ AI ซึ่งถือเป็นส่วนสำคัญอย่างมากในการสร้างความฉลาด โดยกลไกการทำงานของ AI จะต้องอาศัย Machine Learning ที่ประกอบด้วย อัลกอริทึม (Algorithm) หรือ ชุดคำสั่งหรือเงื่อนไขแบบทีละขั้นตอน ที่จะทำให้คอมพิวเตอร์ หุ่นยนต์ เครื่องจักร หรืออุปกรณ์อิเล็กทรอนิกส์ต่าง ๆ ทำการเรียนรู้ด้วยตนเอง โดยการใช้ข้อมูลขนาดใหญ่ (Big Data) ที่มีอยู่ แล้วประมวลผลออกมาเป็นชุดข้อมูลต่าง ๆ ซึ่งการเรียนรู้ของเครื่อง 2. Deep Learning คือ วิธีการเรียนรู้แบบอัตโนมัติด้วยการเลียนแบบการทำงานของโครงข่ายประสาทของมนุษย์โดยนำระบบโครงข่ายประสาทเทียม มาซ้อนกันหลายชั้นและทำการเรียนรู้ ข้อมูลตัวอย่าง ซึ่งข้อมูลจะถูกนำไปใช้ในการตรวจจับรูปแบบหรือจัดหมวดหมู่ข้อมูล โดย Deep Learning การที่เครื่องจะสามารถเข้าใจสิ่งต่าง ๆ ได้จำเป็นที่จะต้องมีองค์ความรู้ก่อน จากนั้นจะประเมินชุดข้อมูลและนำเสนอหรือแทนองค์ความรู้นั้น คำว่า Deep Learning มีที่มาจากการที่ใช้ layer

Computer Science Department
Faculty of Informatics, Mahasarakham University

ของ Neural Network หลาย ๆ อัน มาต่อกัน (มี Hidden Layer มากกว่า 2 Layer ก็ถือเป็น Deep Learning แล้ว) เนื่องจาก layer เหล่านี้เป็นโครงสร้างที่ถูกจัดเก็บแบบเป็นกองซ้อน (Stack) จึงเปรียบได้ว่า Layer ที่จำนวนเยอะ ๆ ก็จะทำให้มีโครงสร้างที่ลึก (Deep)



ภาพประกอบที่ 1 Deep Learning

2.2 CNN (Convolutional Neural Network)

Convolutional Neural Network เป็นโครงข่ายประสาทที่เค้ามีไอเดียในการแก้ปัญหาทางด้านรูปภาพ ซึ่งในเวลานั้นถือเป็นการปฏิวัติวงการ Image Recognition การทำงานแบบคร่าว ๆ มีรูปมาใช้ นำรูปมาเข้ากระบวนการ Convolution เพื่อเพิ่มรายละเอียด เพิ่ม Noise ให้กับรูป Input จากนั้นก็ใส่ Polling เข้าไป (มีหลายแบบ Average, Max) คือ ทำไปเพื่อให้มันช่วยในการหา Pattern ในขั้นตอนต่อไป จากนั้นก็นำ output ที่ได้ ส่งต่อไปให้กับโมเดล Neural Network เพื่อเรียนรู้ปรับ Weight และ ทำให้เกิดกระบวนการสอนให้เรารู้ว่าแต่ละ Combination ของ Feature ที่ถ่วงน้ำหนักมาแล้วนั้นหมายถึงอะไร โดยกระบวนเรียนรู้จะถูกทำซ้ำเพื่อปรับค่าพารามิเตอร์หลาย ๆ รอบ (Epoch) เพื่อให้ค่า Error จากการทำนายลดลงในแต่ละรอบ

2.2.1 การทำงานของ CNN

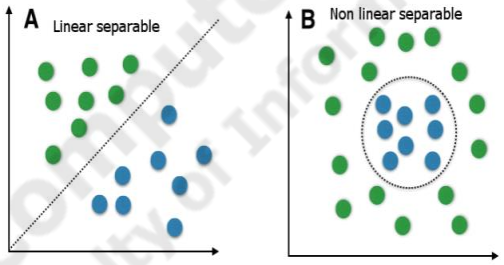
สมมุติ CNN รับรูปภาพ input เป็นรูปแมว รูปภาพเหล่านี้จะถูกจัดเก็บในรูปแบบของ Pixel ซึ่งโดยทั่วไปแล้ว จะใช้ 1 Layer ในการเก็บข้อมูลในรูปแบบเฉดสีขาวดำ (Greyscale) และจะใช้อีก 3 Layer ในการเก็บข้อมูลในรูปแบบเฉดสีต่าง ๆ ระหว่างที่ Model ทำการเรียนรู้ (Learn) (สกัด Feature) ที่ Hidden Layer, Model จะทำการหา Feature ที่มีความเฉพาะเจาะจงต่อข้อมูล input ที่ได้รับ อย่างเช่น ในกรณีนี้ อาจได้เป็น Feature ที่ตรวจสอบหางของแมว เป็นต้น เมื่อ Model ทำการเรียนรู้ (Learn) เสร็จแล้ว มันจะสามารถให้คำตอบแต่ละรูปภาพเป็นความน่าจะเป็น และหากรูปภาพนั้น ๆ มีความน่าจะเป็นแบบใดสูงที่สุด Model จะตอบเป็นสิ่งนั้น เช่น หากโมเดลวิเคราะห์แล้วว่า รูปภาพนั้นมีความน่าจะเป็นว่าจะเป็นแมวสูง Model ก็จะให้คำตอบว่ารูปภาพนั้นคือภาพแมวนั่นเอง

2.3 SVM (Support Vector Machine)

Support Vector Machine (SVM) เป็น Algorithm แบบ Supervised Learning ที่ใช้สำหรับแก้ปัญหาการจัดกลุ่มข้อมูล Classification และการวิเคราะห์การถดถอย Regression ซึ่งจะมีความคล้ายคลึงกับ Logistic Regression (LR) Support Vector Machine จะเป็นการจัดกลุ่มข้อมูล Classification โดยการแบ่ง Class ของข้อมูล

Computer Science Department
Faculty of Informatics, Mahasarakham University

ออกจากกัน ซึ่งสามารถใช้การแบ่งด้วยสมการเชิงเส้นได้ทั้ง Linear และ Non Linear การจัดกลุ่มข้อมูล Classification สามารถแบ่งข้อมูลได้หลาย Model โดยใช้สมการเส้นตรง Linear แต่ปัญหาคือแล้ว Model ไหนที่ดีที่สุด ซึ่ง Support Vector Machine จะใช้ในการหา Hyperplane ซึ่งก็คือ Model ที่ดีที่สุด จากระยะห่างสูงสุดของแต่ละ Class ข้อมูล Maximum Margin ในแต่ละมิติ N – Dimension โดย N เป็นจำนวนของ Feature (2 มิติ หรือ 3 มิติ) Hyperplane จะเกิดจากการลากเส้นผ่าน Support Vector ซึ่งก็คือ Data Point ที่มีระยะห่างของ Maximum Margin มากที่สุดของแต่ละ Class ข้อมูล ซึ่งจะถูกเรียกว่า Decision Line กรณีที่ข้อมูลเป็นแบบ 2 มิติ แต่ในทางคณิตศาสตร์จะถูกเรียกว่า Hyperplane เนื่องจากใน Dimension ที่มากกว่า 2 มิติ จะไม่เป็นเส้นนั่นเอง



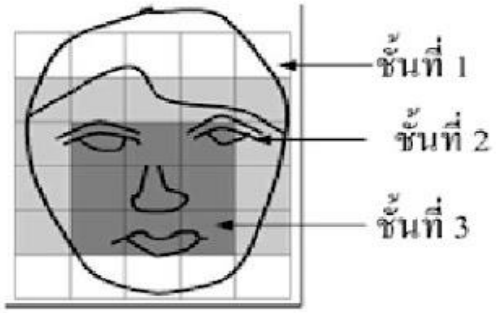
ภาพประกอบที่ 2 SVM Class

2.1.5 ทฤษฎีเกี่ยวกับการตรวจจับใบหน้าและดวงตา

การค้นหาใบหน้าของบุคคลจากภาพนั้นเป็นงานวิจัยที่ได้รับการยอมรับและศึกษาค้นคว้ากันอย่างกว้างขวางเนื่องมาจากการค้นหาใบหน้าเฉพาะของแต่ละบุคคลจำเป็นต้องใช้คุณลักษณะจำเพาะทางสรีระ (Biometric) ที่มีความแตกต่างกัน เช่น โครงหน้า ความกว้างของจมูก และขนาดของโหนกแก้ม เป็นต้น รวมไปถึงปัจจัยภายนอกต่างๆ ที่ส่งผลกระทบต่อการประมวลผลใบหน้า เช่น แสง เงา สี รวมไปถึงกิริยาท่าทางและอารมณ์ที่แสดงออกทางสีหน้าเป็นต้น โดย (Yan, Kriegman and Ahuja, 2022) ได้เสนอการจำแนกประเภทของวิธีการตรวจจับใบหน้า แบ่งออกเป็น 4 ประเภท ดังนี้

2.1.5.1 การใช้กฎเกณฑ์พื้นฐานของมนุษย์ (Knowledge-based Method)

เป็นการพิจารณาหาความสัมพันธ์ระหว่างการใช้ระยะทางและตำแหน่งตามกฎเกณฑ์ที่ตั้งไว้ และองค์ประกอบต่าง ๆ บนใบหน้า ในงานวิจัยของ (Yang and Huang, 1994)และ (Kouropoulos, Pitas, 1997) โดยศึกษาวิธีเชิงความรู้แบบชั้น (Hierarchical) และวิธีฐานกฎแบบกำหนดขอบเขต (Rule-based localization) ตามลำดับ ดังภาพประกอบที่ 3



ภาพประกอบที่ 3 ตำแหน่งความน่าจะเป็นภาพรวมของใบหน้า

จากภาพประกอบที่ 3 แสดงให้เห็นถึงการพิจารณาขอบเขตของแต่ละลำดับชั้น ในชั้น

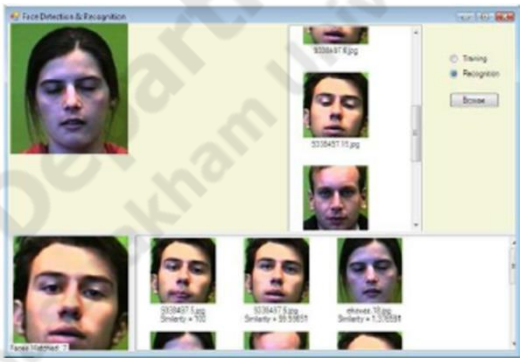
ที่ 1 พิจารณาหาตำแหน่งความน่าจะเป็นภาพรวมของใบหน้า ชั้นที่ 2 พิจารณาหาค่าฮิสโทแกรมบนใบหน้า (Histogram) และเส้นขอบ ชั้นที่ 3 พิจารณาลักษณะเด่นบนใบหน้า จากความสมมาตรขององค์ประกอบบนใบหน้าในแนวนอนและแนวตั้ง เช่น จมูก ดวงตา และปาก

ต่อมา Kouropoulos and Pitas (1997) ได้ศึกษาวิธีฐานกฎแบบกำหนดขอบเขต (Rule-based localization) ซึ่งมีความคล้ายคลึงกับวิธีของ YangandHuang (1994) แต่แตกต่างกันที่ค่าฮิสโทแกรม(Histogram)ในตำแหน่งแนวตั้งและแนวนอนของภาพ ดังสมการต่อไปนี้

$$HI(x) = \sum_{y=1}^{n} I(x,y)$$

$$HI(y) = \sum_{x=1}^{n} I(x,y)$$

นอกจากนี้ยังพบอุปสรรคคือการหากฎเกณฑ์เฉพาะที่ใช้ในการจำแนกใบหน้าเนื่องจากมีรายละเอียดซับซ้อนและมากเกินไปส่งผลให้ตรวจจับใบหน้าได้ยาก(ไม่มีใบหน้าใดผ่านเกณฑ์)หรือหากมีรายละเอียดที่น้อยจนเกินไปส่งผลให้เกิดข้อผิดพลาดในการตรวจจับใบหน้า (เกิดการตรวจจับสิ่งของ) นอกจากนี้ยังไม่สามารถตรวจจับภาพที่เคลื่อนไหวได้ เพราะมีความซับซ้อน และมีตำแหน่งไม่คงที่ของใบหน้า การค้นหาลักษณะเด่น (Feature Invariant Method) เป็นการใช้อัลกอริทึมวิเคราะห์ใบหน้าเพื่อหาลักษณะเด่น (Feature) และโครงสร้างของใบหน้า ดังภาพที่ 2.6 รวมไปถึงใช้การเปลี่ยนแปลงองค์ประกอบของภาพเข้ามาเกี่ยวข้อง เช่น แสง เงา เป็นแบบจำลองในการตรวจพบใบหน้า โดยแบบจำลองประกอบด้วยจุดความสว่างน้อยเพื่อแสดงดวงตา และจุดความสว่างมากแสดงโหนกแก้มและจมูก หลังจากนั้นจึงหาความสัมพันธ์ของระยะห่างและพิกัดของจุดต่าง ๆ เพื่อคัดเลือกใบหน้าที่เหมาะสม



ภาพประกอบที่ 4 ตัวอย่างใบหน้า

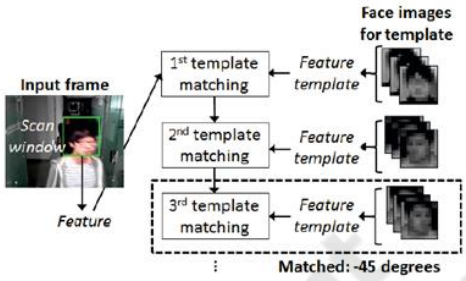
จากงานวิจัยของ (Yow and Cipolla, 1996) ได้นำลักษณะเด่นของใบหน้ามาวิเคราะห์ โดยการตรวจจับขอบภาพ (Edge Detection) ดังภาพที่ 15 จากนั้นใช้แบบจำลองทางสถิติ (Statistical Model) อธิบายความสัมพันธ์และการยืนยันการค้นพบใบหน้า ในทางตรงกันข้ามหากองค์ประกอบของภาพไม่คงที่ ส่งผลต่อการค้นหาใบหน้าของระบบ

Computer Science Department
Faculty of Informatics, Mahasarakham University

ภาพประกอบที่ 5 HOG

2.1.5.3 การใช้ต้นแบบมาตรฐาน (Template Matching Method)

เป็นการใช้ต้นแบบมาตรฐานของใบหน้าที่ถูกสร้างขึ้นเองด้วยมือ (Manual) หรือถูกกำหนดขึ้นด้วยฟังก์ชันทางคณิตศาสตร์ (Mathematical Function) เพื่อวิเคราะห์องค์ประกอบของใบหน้า จากงานวิจัยของ (Scassellati, 1998) ได้ทำการศึกษาระบบตรวจจับใบหน้า โดยกำหนดต้นแบบของใบหน้าที่มีองค์ประกอบ 16 พื้นที่ และ 23 ความสัมพันธ์ซึ่งข้อมูลภาพที่รับเข้ามาจะถูกนำมาวิเคราะห์ หาค่าสหสัมพันธ์ (Correlation Value) กับใบหน้าที่เป็นต้นแบบมาตรฐาน (Standard Pattern) ในส่วนองค์ประกอบบนใบหน้า จมูก ปาก และดวงตา อย่างเป็นอิสระต่อกัน ดังภาพประกอบที่ 2.8 ซึ่งผลลัพธ์ที่ปรากฏบนใบหน้าหาค่าได้จากค่าสหพันธ์เป็นส่วนใหญ่



ภาพประกอบที่ 6 การหา feature

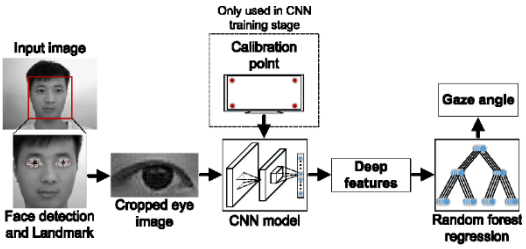
จากภาพประกอบที่ 2.8 เป็นการใช้ภาพต้นแบบมาตรฐานของใบหน้าที่ถูกกำหนดไว้เพื่อค้นหาใบหน้าตามความสัมพันธ์ระหว่างภาพต้นแบบมาตรฐาน และข้อมูลภาพที่นำเข้า ซึ่งเป็นวิธีที่สามารถนำไปใช้งานได้ง่าย แต่มีประสิทธิภาพในการตรวจจับใบหน้าไม่ดีพอ เนื่องจาก ผลลัพธ์ของการวางตำแหน่ง ขนาดภาพ และรูปทรงใบหน้าที่แตกต่างจากต้นแบบมาตรฐาน

2.1.5.4 การใช้รูปลักษณ์ภายนอก (Appearance-Based Method)

เป็นการใช้แบบจำลองที่ได้จากการศึกษาของกลุ่มภาพตัวอย่างที่เป็นใบหน้า และไม่ใช่ใบหน้า โดยทั่วไปแล้วการสร้างแบบจำลองดังกล่าวจะอยู่บนพื้นฐานทางเทคนิคการวิเคราะห์สถิติ (Statistical-based Method) และการเรียนรู้ของเครื่องกล (Machine Learning) เพื่อค้นหาคุณลักษณะความสัมพันธ์ของรูปร่างใบหน้า และที่ไม่ใช่รูปร่างใบหน้า ดังภาพที่ 17 ซึ่งคุณลักษณะที่ได้รับจากการศึกษาจะถูกจัดอยู่ในรูปแบบจำลองการกระจาย (Distribution Models) หรือฟังก์ชันการจำแนก (Discriminant Functions) นอกจากนี้

Computer Science Department
Faculty of Informatics, Mahasarakham University

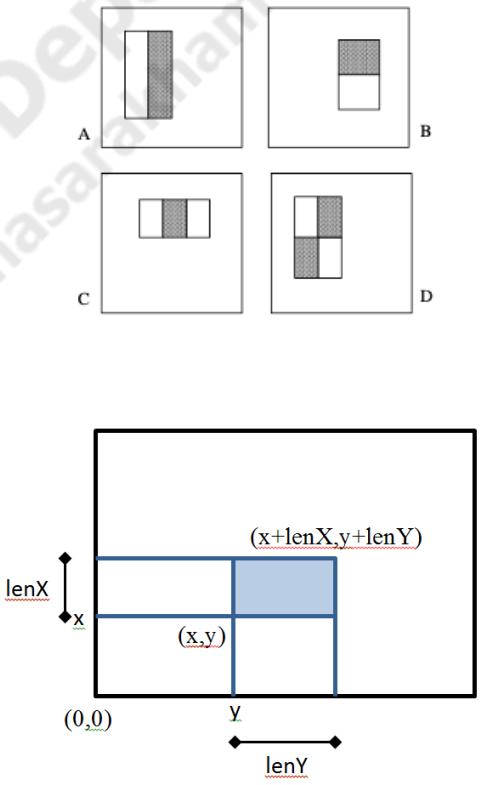
ยังใช้การลดมิติ (Dimension Reduction) เพื่อเพิ่มประสิทธิภาพการคำนวณและการตรวจจับใบหน้า ซึ่งจากงานวิจัยของ (Sung และ Pogio, 1998) ได้ใช้การประมาณฟังก์ชันความหนาแน่นของรูปใบหน้า และไม่ใช่ใบหน้าด้วยการใช้กลุ่มของเกาส์เซียน (Gaussian)



ภาพประกอบที่ 7 การค้นหาใบหน้า

วิธีการตรวจจับใบหน้าของ Viola-Jones ได้เสนอเทคนิคการตรวจจับใบหน้าที่มีความเร็วและมีความถูกต้องในการตรวจจับสูง ซึ่งกลายเป็นเทคนิคที่ได้รับการยอมรับและเผยแพร่ในงานวิจัยการตรวจจับใบหน้าอย่างแพร่หลาย โดยหลักการพื้นฐานของเทคนิคการตรวจจับใบหน้าของ Viola-Jones คือ การนำภาพที่ตรวจหาใบหน้ามาแบ่งเป็นภาพย่อย (Sub-window) ซึ่งภาพย่อยจะถูกนำมาเป็นภาพนำเข้าของกระบวนการตรวจหาใบหน้าโดยการปรับขนาดของภาพให้แตกต่างกันหลาย ๆ ขนาด และใช้เครื่องตรวจจับ (Detector) ขนาดคงที่ค้นหาวัตถุ โดยเทคนิคการตรวจจับใบหน้าของ Viola-Jones สามารถแบ่งออกได้ 3 ขั้นตอน ดังนี้

1.) การคำนวณรูปแบบการจำลองด้วย Integral Image Viola-Jones เสนอให้ใช้การคำนวณแบบ Integral Image หาค่าในการจำลองรูปแบบ Hear-like คือ การหาผลต่างระหว่างความเข้ม (Intensity) ในส่วนที่แรเงาและไม่ได้แรเงา จากนั้นจึงนำคำนวณหาความเข้มของพื้นที่แรเงา ดังภาพที่ 2.7 เพื่อเพิ่มประสิทธิภาพในการตรวจจับใบหน้า เมื่อ ii (x, y) คือ ค่าของ Integral image ที่ตำแหน่งจุด (x, y) และ (x', y') คือ ค่าความเข้ม ในแต่ละส่วนที่แรเงาของภาพต้นฉบับ ดังภาพประกอบที่ 2.10

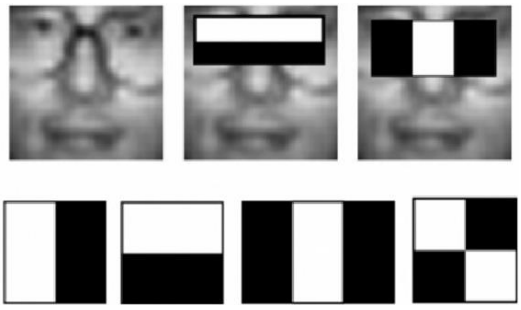


ภาพประกอบที่ 9 จำลองรูปแบบ Hear-like

2.) ขั้นตอนการเรียนรู้ด้วยวิธี Adaboost เป็นการนำแบบจำลองรูปแบบ Hear-like จากขั้นตอนแรกมาเข้ากระบวนการเรียนรู้ของเครื่อง Machine Learning ซึ่ง

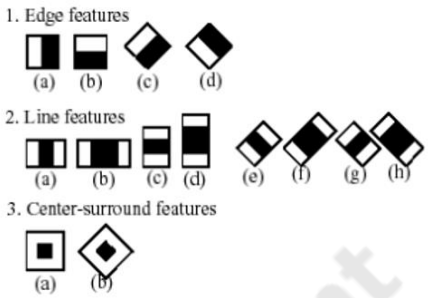
Computer Science Department Faculty of Informatics, Mahasarakham University

เรียกว่า "Adaptive Boost" หรือ "Adaboost" ใช้เร่งหาตัวจำแนกแบบอ่อนแอ และตั้งค่าถ่วงน้ำหนักที่ทำให้เกิดค่าผิดพลาดน้อยที่สุดในแต่ละรอบของกระบวนการ เพื่อสร้างตัวจำแนกแบบแข็งแรง ดังภาพประกอบที่ 2.12



ภาพประกอบที่ 10 จำลองรูปแบบ Hear-like

3.) การรวมตัวจำแนกกลุ่มแบบต่อเรียง (Cascaded Classifier) เป็นการสร้างการรวมตัวจำแนกกลุ่มแบบต่อเรียง (Cascaded Classifier) โดยนำตัวจำแนก (Classifier) หลาย ๆ ตัวต่อกันตามลำดับ ดังภาพที่ 2.9 ซึ่งตัวจำแนกในลำดับต้น ๆ จะมีความซับซ้อนน้อยกว่า โดยสันนิษฐานว่ามาจากตัวจำแนกแบบอ่อนแอ เพื่อลดปริมาณการคำนวณและลดระดับค่าแบ่งขีด (Threshold) เพื่อลดความผิดพลาดในการตรวจจับไม่พบใบหน้า และภาพย่อยที่ไม่ใช่ใบหน้า (Non-face) จะถูกตัดทิ้งทันทีเป็นจำนวนมาก ในทางตรงกันข้ามภาพย่อยที่อาจจะเป็นใบหน้า จะถูกส่งต่อไปยังตัวจำแนกถัดไปที่ซับซ้อนขึ้นเรื่อย ๆ ซึ่งถ้ามีจำนวนชั้นของตัวจำแนกมาก จะส่งผลให้เกิดภาพย่อยที่เป็นใบหน้ามากยิ่งขึ้น



ภาพประกอบที่ 11 Cascaded Classifier

2.1.7 OpenCV

OpenCV (Open source Computer Vision) เป็นไลบรารีฟังก์ชันการเขียนโปรแกรม (Library of Programming Functions) โดยส่วนใหญ่จะมุ่งเป้าไปที่การแสดงผลด้วยคอมพิวเตอร์แบบเรียลไทม์ (Real-Time Computer Vision) เดิมทีแล้วถูกพัฒนาโดย Intel แต่ภายหลังได้รับการสนับสนุนโดย Willow Garage ตามมาด้วย Itseez (ซึ่งต่อมาถูกเข้าซื้อโดย Intel) OpenCV เป็นไลบรารีแบบข้ามแพลตฟอร์ม (Cross-Platform) และใช้งานได้ฟรีภายใต้ลิขสิทธิ์ของ BSD แบบโอเพ่นซอร์ส (Open-Source BSD License) OpenCV ยังสนับสนุนเฟรมเวิร์กการเรียนรู้เชิงลึก (Deep Learning Frameworks) ได้แก่ TensorFlow, Torch/PyTorch และ Caffe ตัวอย่างการประยุกต์ใช้งาน OpenCV มีดังนี้

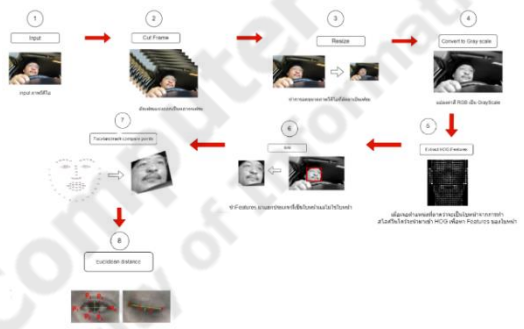
- ชุดเครื่องมือคุณลักษณะ 2 มิติและ 3 มิติ (2D and 3D feature toolkits)
- การประมาณระยะในขณะเคลื่อนที่ (Egomotion Estimation)

Computer Science Department Faculty of Informatics, Mahasarakham University

- ระบบรู้จำใบหน้า (Facial recognition system)
- การจดจำท่าทาง (Gesture recognition)
- ปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์ (Human-Computer interaction; HCI)

ภาษาการเขียนโปรแกรม OpenCV ถูกเขียนขึ้นด้วยภาษา C++ มีการรองรับ Python, Java และ MATLAB/OCTAVE—API สำหรับอินเทอร์เฟสเหล่านี้สามารถพบได้ในเอกสารออนไลน์ ซึ่งมีการรวมไว้หลากหลายภาษา เช่น C#, Perl, Ch, Haskell และ Ruby ได้รับการพัฒนาเพื่อส่งเสริมการนำมาใช้งานโดยผู้ใช้ที่เพิ่มขึ้น

3.แผนการดำเนินงาน



ภาพประกอบที่ 12 แผนการดำเนินงาน

4.ผลการทดลอง

4.1 วิธีการทดลอง

ในการเตรียมชุดข้อมูลเพื่อทดลองและวัดประสิทธิภาพ ชุดข้อมูลประกอบด้วยทั้งหมด 800 ภาพ ซึ่งถูกสุ่มเลือกมาจากวิดีโอทั้งหมด 20 วีดีโอ โดยแต่ละวิดีโอมีความยาว 1 – 2 นาที ไม่เท่ากันและทำการตัดออกทุก ๆ 30 เฟรม เพื่อให้มีความห่างกันและแบ่งเป็นภาพที่ตัดออกมาเป็น 3 ส่วน ภาพที่ใบหน้าที่ก้มหน้าจำนวน 10 ภาพ/วิดีโอ ภาพใบหน้าที่หันซ้ายและขวาจำนวน 10 ภาพ/วิดีโอ และภาพใบหน้าที่ตรงปกติจำนวน 20 ภาพ/วิดีโอ

4.2 ผลการทดลองการหาใบหน้า

ผลการทดลองการค้นหาใบหน้าโดยนำภาพที่ตัดออกมาและสุ่มมาจากวิดีโอจำนวน 20 วิดีโอและภาพที่ตัดออกมาทั้งหมด 800 ภาพ เป็นภาพนิ่งทั้งหมดมาเข้าทดลองในโปรแกรมเพื่อให้โปรแกรมตรวจหาใบหน้า โดยจำนวนใบหน้าก้ม 10 ภาพ จำนวนใบหน้าหันซ้ายขวา 10 ภาพ และจำนวนใบหน้าตรงทั้งหมด 20 ภาพ รวม หนึ่งวิดีโอทั้งหมดจำนวน 40 ภาพ

ข้อจำกัดของการใช้โปรแกรมเพื่อหาใบหน้าจากภาพ ภาพที่สามารถหาว่าเป็นใบหน้าได้อย่างถูกต้องและมีความแม่นยำ นั้นจะต้องเห็นดวงตาชัดเจนทั้งสองข้าง เพราะถ้าไม่เห็นดวงตาบนใบหน้าชัดทั้งสองข้าง หรือเห็นแค่หนึ่งข้าง Facial Landmarks จะไม่นับว่าเป็นใบหน้า Facial Landmarks คือ จุดสำคัญบนใบหน้า เช่น ตา คิ้ว จมูก ปาก ที่จะใช้หาใบหน้า ตัวอย่างภาพที่ตรวจจับว่าเป็นใบหน้า ภาพ ก ที่ตรวจว่าเป็นใบหน้าเพราะเห็นดวงตาครบทั้งสองข้างโปรแกรมจึงบอกว่าเป็นใบหน้า ตัวอย่างภาพที่ไม่สามารถตรวจจับใบหน้าได้จะ

Computer Science Department Faculty of Informatics, Mahasarakham University

เห็นว่าภาพ จ เป็นการหันหน้าของผู้ขับขี่จึงทำให้เห็นดวงตาฝั่งซ้ายของผู้ขับขี่ไม่ชัดเจน ทำให้โปรแกรมตรวจไม่เจอ Facial Landmarks ของดวงตาและภาพ ฉ เป็นการก้มหน้าของผู้ขับขี่ทำให้มองไม่เห็นดวงตาทั้งสองข้าง โปรแกรมจึงบอกว่าไม่ใช่ใบหน้า

4.3 ผลการทดลองการตรวจจับการหลับตา (จากรูปภาพ)

นำภาพที่โปรแกรมตรวจจับว่าเป็นใบหน้าได้ทั้งหมด จากผลการทดลองที่ 4.2 การค้นหาใบหน้า ทั้งหมด 683 ภาพที่จำแนกว่าเป็นใบหน้า มาทดสอบการตรวจจับการหลับตาโดยใช้โมเดลเข้ามาช่วยในขั้นตอนการตรวจจับการหลับตาจากภาพนิ่ง ชุดข้อมูลนำเข้าถูกนำเข้าไปยังโมเดลการตรวจจับ เพื่อวัดประสิทธิภาพในการตรวจจับการหลับตา ผลการทดลองแสดงดังตารางที่ 4.3 ซึ่งจะประกอบด้วยค่า TP TN FN และ FP เพื่อจะนำไปหาค่าความถูกต้อง (Accuracy) ค่า Precision ค่า Recall และค่า TPR, FPR เพื่อใช้ในการวัดประสิทธิภาพโมเดล

จากการทดลองพบว่า มีภาพใบหน้าที่สามารถตรวจจับได้ว่าเป็นภาพที่หลับตาจำนวน 141 ภาพ ตรวจจับได้ว่าเป็นภาพที่หลับตาจำนวน 308 ภาพ ตรวจจับได้ว่าไม่หลับแต่หลับตาจำนวน 100 ภาพและตรวจจับได้ว่าหลับแต่บอกว่าไม่หลับตาจำนวน 28 ภาพ คิดเป็นเปอร์เซ็นความถูกต้อง 77.81 %และข้อจำกัดที่ทำให้ไม่สามารถทำให้ตรวจจับว่าเป็นใบหน้าและหลับตาได้ มีผลมาจากการก้มหน้ามากเกินทำให้โปรแกรมเห็นดวงตาไม่ชัดเจนและหันซ้ายขวาจนโปรแกรมไม่เห็นดวงตาทั้งสองข้างแบบชัดเจน หรือเห็นข้างเดียวจะไม่นับว่าภาพนั้นเป็นใบหน้าและหลับตา ดังตารางที่ 4.4 การวัดประสิทธิภาพการตรวจจับการหลับตา

4.4 การทดลองการตรวจจับภาวะหลับใน (จากวิดีโอ)

ในการวัดประสิทธิภาพเพื่อตรวจจับภาวะหลับในนั้น เราจะใช้โมเดลในการตรวจจับภาวะหลับใน ในขั้นตอนนี้เราจะนำวิดีโอทั้งหมด 20 วิดีโอที่มีความยาว 2 – 3 นาที ไม่เท่ากันมาเข้าเพื่อทดสอบประสิทธิภาพของโปรแกรมและโมเดล โดยให้โมเดลตรวจว่ามีช่วงการหลับตามากว่า 1 - 2 วินาทีขึ้นไป โปรแกรมจะส่งสัญญาณว่า ช่วงนั้นผู้ขับขี่ได้เข้าสู่ภาวะหลับใน เรายกตัวอย่างมา 15 ช่วงเพื่อทดสอบประสิทธิภาพ มีจำนวน 20 ตาราง

จากการทดลองนำวิดีโอทั้ง 20 วิดีโอเข้าทดสอบในโปรแกรมเพื่อวัดประสิทธิภาพของโปรแกรมนั้นพบว่า มีช่วงที่สามารถตรวจจับว่าหลับในจริงทั้งหมด 59 ช่วง ช่วงที่ไม่หลับจริงทั้งหมด 167 ช่วง ช่วงที่หลับแล้วโปรแกรมตอบว่าไม่หลับทั้งหมด 111 ช่วงและช่วงที่ไม่หลับแล้วโปรแกรมตอบว่าหลับทั้งหมด 39 ช่วง คิดเป็นร้อยละความถูกต้องของระบบทั้งหมดจะได้ 61.57 % ร้อยละความถูกต้องของการตรวจจับภาวะหลับในจะได้ 34.70%

Computer Science Department
Faculty of Informatics, Mahasarakham University

และร้อยละความถูกต้องของการตรวจจับภาวะปกติจะได้ 83 % และข้อจำกัดที่ทำให้โปรแกรมไม่สามารถตรวจจับภาวะหลับในได้ถูกต้องทั้งหมดมาจากโปรแกรมเห็นดวงตาของผู้ขับขี่ทั้งสองข้างหรือเห็นดวงตาไม่ชัดเจน ซึ่งเกิดมาจากการที่ผู้ขับขี่ก้มหน้ามากเกินไปและหันและศีรษะจนไม่เห็นดวงตาทั้งสองข้าง

4.5 การทดลองการตรวจจับภาวะหลับในจากวิดีโอ (ใช้โมเดลที่ผู้พัฒนาฝึกฝน)

4.5.1 การเตรียมชุดข้อมูลก่อนการเรียนรู้

ในการเตรียมชุดข้อมูลจะแบ่งเป็นข้อมูลรูปภาพที่ใช้ในการเรียนรู้ (Train) และใช้ทดสอบความถูกต้อง (Validation) จากรูปภาพทั้งหมด 1,483 ภาพ แบ่งชุดข้อมูลดังนี้

1.จำนวนข้อมูลสำหรับการเรียนรู้ใช้ข้อมูลภาพ 1,000 ภาพ คิดเป็นประมาณ (67%)

2.จำนวนข้อมูลที่ใช้ในการตรวจสอบความถูกต้องใช้ข้อมูลภาพ 483 ภาพ คิดเป็นประมาณ (33%)

4.5.2 การตั้งค่าการฝึกฝนเรียนรู้

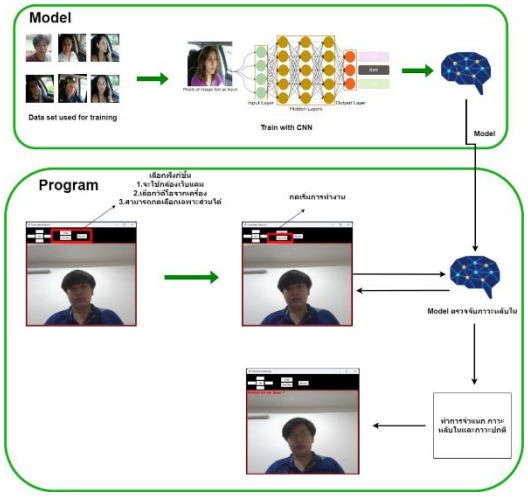
ตารางที่ 1 การตั้งค่าการฝึกฝนเรียนรู้

Function	Description
Epoch กำหนดรอบการเรียนรู้	30
Batch size ขนาดของชุดข้อมูล	50
Image size กำหนดขนาดภาพที่จะนำมาทำการฝึกฝน	224
Activation for Multi class แปลงผลรวมเพื่อให้ได้ผลลัพธ์สำหรับการจัดหมวดหมู่หลับและไม่หลับ	sigmoid
Optimizer	RMSprop, Learning rate 0.01

4.5.3 วิธีการทดสอบ

โดยการทดสอบโมเดลจะใช้วิดีโอจำนวน 20 วิดีโอในการทดสอบโมเดล วิดีโอจะมีความยาว 2 – 3 นาที และมีผู้ขับขี่อยู่ในรถที่แตกต่างกันมีทั้งช่วงที่เกิดภาวะหลับในและไม่เกิดภาวะหลับใน โดยมีกระบวนการนำวิดีโอเข้าทดสอบ ดังนี้

Computer Science Department
Faculty of Informatics, Mahasarakham University

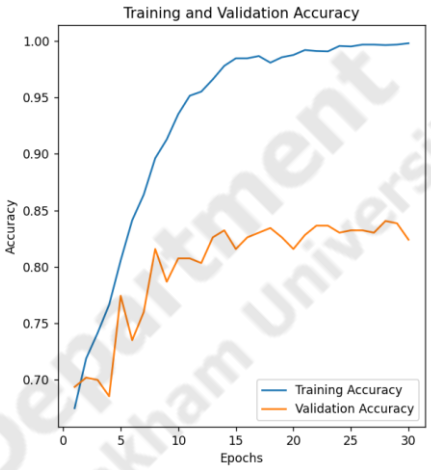


ภาพประกอบที่ 13 การทำงานของโมเดล

4.5.4 ผลการทดลองและประเมินประสิทธิภาพโมเดล

ในการทดสอบและประเมินประสิทธิภาพของโมเดล เพื่อช่วยจำแนกและพยากรณ์ภาวะหลับในในเวลาขับรถโดยใช้สถาปัตยกรรมโมเดล Convolutional Neural Network (CNN) และใช้ Validation Loss ในการประเมิน รูปแบบการคำนวนค่า Loss แบบ binary crossentropy และได้เลือกใช้ Optimizer รูปแบบ RMSprop ซึ่งผลลัพธ์ของค่า Loss ที่ได้ระหว่างการเรียนรู้สามารถอธิบายได้ดังนี้

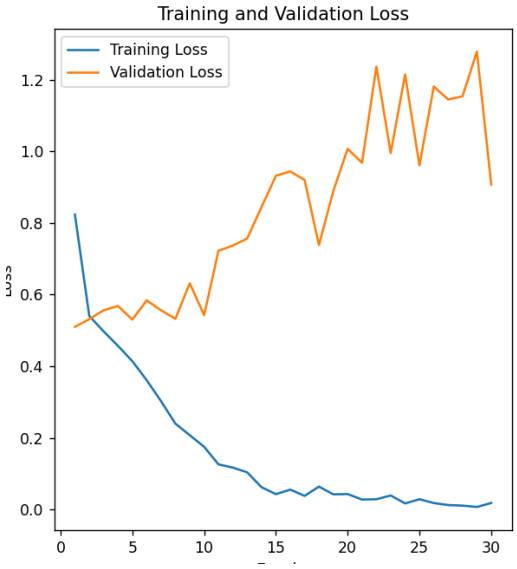
ผลลัพธ์กราฟที่ 1 จากการฝึกฝนโมเดล 30 รอบ ด้วยชุดข้อมูลสำหรับใช้ข้อมูลในการฝึกฝนเรียนรู้ 1,000 ภาพ และภาพที่ใช้ตรวจสอบความถูกต้อง 483 ภาพ ซึ่งผลการทดสอบจากการเรียนรู้ 30 รอบ จะเห็นได้ชัดเจนเลยว่าเส้น Training Accuracy เพิ่มขึ้นเรื่อย ๆ และไปจบที่ 0.9980 ส่วนเส้นของ Validation ในช่วงรอบที่ 3 – 4 จะมีการลดลงและไม่มีแนวที่จะเพิ่มขึ้นไปมากกว่านี้ได้ ค่าของ Validation จะอยู่ที่ 0.8240



ภาพประกอบที่ 14 ค่าความถูกต้องของโมเดล

ผลลัพธ์กราฟที่ 2 แสดงให้เห็นถึงค่า Loss หรือค่าความผิดพลาดที่เกิดขึ้นจากการเรียนรู้ของโมเดลทั้ง 30 รอบ ซึ่งอธิบายได้ว่าการเรียรรู้ในแต่ละรอบนั้นทำให้โมเดลมีความแม่นยำเพิ่มขึ้นจะเห็นได้ว่า เส้น Training Loss นั้นมีค่า Loss หรือค่าความผิดพลาดเข้าใกล้ 0 ประมาณ 0.018 และไม่สามารถไปต่อได้อีก ส่วนค่า Validation Loss มีค่าความผิดพลาดเพิ่มขึ้นในช่วงที่ 18 – 22 และค่อย ๆ ลดลงในรอบที่ 29

Computer Science Department
Faculty of Informatics, Mahasarakham University



ภาพประกอบที่ 15 ค่าความผิดพลาดของโมเดล

4.5.5 การประเมินและวิเคราะห์ผลการประเมิน

การประเมินผลจากการวัดประสิทธิภาพจากการใช้วิดีโอทั้งหมด 20 วิดีโอในการวัดประสิทธิภาพของโมเดลเพื่อหาค่าความถูกต้อง (Accuracy) ค่าความแม่นยำ (Precision) และ Recall ผลลัพธ์ที่ได้จากการทดสอบการตรวจจับภาวะหลับในเพื่อจำแนกภาวะหลับในจากการทำนายผลของโมเดล

ตารางที่ 2 ตารางวัดประสิทธิภาพ

ผลการวัดประสิทธิภาพของโปรแกรม		
Accuracy	Recall	Precision
0.50	0.54	0.26

4.6 สรุปผลการทดลอง

จากการทดลองในการเปรียบเทียบประสิทธิภาพโมเดลระหว่างโมเดลที่ 1 และโมเดลที่ 2 พบว่าโมเดลที่ 1 มีประสิทธิภาพที่ดีที่สุดในการใช้งานตรวจจับภาวะหลับใน และข้อจำกัดของโปรแกรมในการตรวจจับภาวะหลับในที่ไม่สามารถตรวจจับได้บางช่วง เนื่องจากในช่วงที่เกิดภาวะหลับในมีสถานการณ์ที่เกิดขึ้น เช่น ผงกศีรษะเร็วจนโปรแกรมตรวจไม่เจอใบหน้า หลับตาไม่ถึง 1 วินาทีแล้วผงกศีรษะเร็วทำให้โปรแกรมมองว่าผู้ขับขี่นั้นตื่นอยู่

ในการทดลองนี้เราใช้โมเดลที่แตกต่างกัน โมเดลที่ 1 ใช้ shape predictor 69 landmarks.dat

ในการช่วยหาจุดบนใบหน้าเพื่อหาดวงตา การตรวจจับภาวะหลับในจะใช้การหาระยะห่างระหว่างขอบตาบนและขอบตาล่าง โมเดลที่ 2 จะเป็นโมเดลที่ทางผู้พัฒนาได้ฝึกฝนเทรนขึ้นมาจะใช้ ในการตรวจจับภาวะหลับในโดยการให้โมเดลทำนายผลออกมา ทั้งสองโมเดลยังมีข้อจำกัดในการที่ทำให้ตรวจจับผิดพลาด ถ้าไม่เห็นดวงตาหรือผงกศีรษะเร็วจนโปรแกรมตรวจไม่เจอใบหน้าโปรแกรมจะไม่ทำการตรวจจับภาวะหลับใน

ตารางที่ 3 ตารางวัดประสิทธิภาพการตรวจจับภาวะหลับใน

Computer Science Department
Faculty of Informatics, Mahasarakham University

ตารางผลการทดลอง					
โมเดลที่ใช้	**TP**	**FP**	**TN**	**FN**	**ร้อยละ**
ผลการทดลองจากโมเดล shape_predictor landmarks.dat	59	112	183	39	61.57 %
ผลการทดลองจากโมเดลที่เทรนเอง	**45**	128	124	38	50.44 %

5.สรุปผลและอภิปรายผลการทดลอง

5.1 สรุปผลและอภิปราย

โครงการปริญญานิพนธ์ฉบับนี้นำเสนอโปรแกรมตรวจจับภาวะหลับใน สำหรับตรวจจับอาการหลับในของผู้ขับขี่รถยนต์ ซึ่งใช้กระบวนการการประมวลผลภาพการเรียนรู้เชิงลึกและโครงข่ายประสาทเทียม มีเป้าหมายเพื่อลดอัตราการเกิดอุบัติเหตุบนท้องถนนที่เกิดจากภาวะหลับในทำให้เกิดการสูญเสีย และสามารถใช้งานผ่านคอมพิวเตอร์ได้ จะแบ่งการทดลองเป็น 2 วิธีดังนี้

การทดลองโกยใช้โมเดล shape_predictor_68_face_landmarks.dat เพื่อตรวจจับใบหน้าและหาดวงตา โดยนำดวงตาตรวจจับอาการหลับหรือปกติ โดยการดูระยะห่างระหว่างตาชั้นบนและชั้นล่างด้วยพีทาโกรัส รายละเอียดการทดลองแสดงในหัวข้อที่ 4.1-4.4 และสรุปได้ดังนี้

1.การตรวจจับอาการหลับ: ตรวจจับได้ทั้งหมด 59 ครั้งจาก 170 ครั้งที่ทดลอง (ความถูกต้อง 34%)

2.การตรวจจับอาการปกติ: ตรวจจับได้ทั้งหมด 183 ครั้งจาก 221 ครั้งที่ทดลอง (ความถูกต้อง 82%)

3.ความถูกต้องรวม: 61.57%

อุปสรรคในการใช้วิธีดังกล่าว คือแสงตกกระทบใบหน้าจากด้านข้าง ของเปลือกตาอาจเกิดเงาและโปรแกรมอาจจับผิดว่าเปลือกตาคือดวงตา

การทดลองโดยใช้ shape_predictor_landmarks เพื่อตรวจจับใบหน้า จากนั้นสร้างโมเดลเพื่อเรียนรู้อาการหลับในและอาการปกติด้วยคอนโวลูชันนิวรอนเน็ตเวิร์ก (CNN) รายละเอียการทดลองแสดงในหัวข้อที่ 4.5 สามารถสรุปผลการทดลองได้ดังนี้

1.การตรวจจับภาวะหลับ: ตรวจจับได้ทั้งหมด 45 ช่วงจาก 170 ช่วงที่ทดลอง (ความถูกต้อง 20%)

2.การตรวจจับภาวะปกติ: ตรวจจับได้ทั้งหมด 124 ช่วงจาก 221 ช่วงที่ทดลอง (ความถูกต้อง 56%)

3.ความถูกต้องรวม: 50.44%

จากการใช้วิธีที่สองมีความถูกต้องน้อยเพราะข้อมูลในการเรียนรู้มีจำนวนน้อย เนื่องจากใช้จำนวน 500 ภาพ สำหรับการเรียนรู้การหลับตาและ 500 ภาพ สำหรับการเรียนรู้ดวงตาเปิดปกติ

5.2 ปัญหาและอุปสรรคในการดำเนินงาน

Computer Science Department
Faculty of Informatics, Mahasarakham University

5.2.1 ในการหาชุดข้อมูลมาทดสอบโมเดลต้องเป็นวิดีโอเท่านั้น ในการหาวิดีโอไม่สามารถหาจากอินเทอร์เน็ตได้หรือหาได้ยาก ผู้พัฒนาจึงต้องทำการถ่ายขึ้นมาเองบางส่วนเพื่อให้มีข้อมูลในการทดลองที่เยอะขึ้น

5.3 ข้อเสนอแนะ

5.3.1 การค้นหาใบหน้าและดวงตาด้วยวิธี shape_predictor_landmarks มีความถูกต้องน้อยในกรณีที่บุคคลมีสีผิวขาวมาก ควรมีการปรับปรุงวิธีเพื่อให้มีความถูกต้องกับบุคคลที่มีสีผิวดังกล่าว

5.3.2 การตรวจสอบบุคคลหลับตาและตาปกติมีจำนวนข้อมูลน้อยจึงควรเพิ่มจำนวนเพื่อให้มีการเรียนรู้ที่มีความถูกต้องเพิ่มขึ้น

อ้างอิง

1. ปัญญาประดิษฐ์ (AI) จาก https://www.etda.or.th/th/Useful-Resource
2. Deep Learning & Machine Learning จาก https://aws.amazon.com/th/what-is/neural-network/
3. CNN (Convolutional neural network) จาก https://medium.com/@natthawatphongchit
4. (Support Vector Machine) & Kernel จาก https://research.psru.ac.th/files/res_che2553/
5. ทฤษฎีการตรวจจับใบหน้า และดวงตา จาก http://sutir.sut.ac.th:8080/jspui/bitstream/
6. การใช้ต้นแบบมาตรฐาน เพื่อวิเคราะห์ใบหน้า จาก http://sutir.sut.ac.th:8080/jspui/bitstream/
7. OpenCV คืออะไร จาก https://medium.com/@nut.ch40/opencv-

Computer Science Department
Faculty of Informatics, Mahasarakham University