บทที่ 2

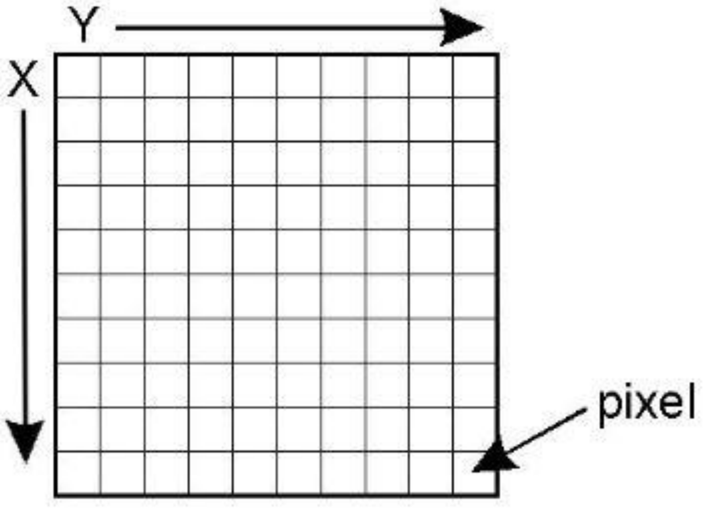
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1. ความรู้พื้นฐานเกี่ยวกับประมวลภาพดิจิทัล

ในปัญหาพิเศษนี้จะใช้ภาพดิจิทัลที่ได้จากกล้องเว็บแคมเป็นข้อมูลรับเข้าของระบบ เพื่อนำภาพเหล่านี้ไปวิเคราะห์ตรวจจับและติดตามลักษณะเด่นตามที่ต้องการ ดังนั้นในหัวข้อนี้จะอธิบายเกี่ยวกับความรู้พื้นฐานเกี่ยวกับการวิเคราะห์ประมวลผลภาพ

2.1.1. ความรู้พื้นฐานเกี่ยวกับภาพดิจิทัล

ภาพดิจิทัลนั้นคือ ภาพที่ถูกแปลงสัญญาณจากรูปแบบแอนะล็อก (Analog) ให้อยู่ในรูปของดิจิทัล (Digital) โดยหลังจากการทำเป็นภาพดิจิทัลแล้ว ภาพเหล่านี้จะถูกลงในหน่วยความคอมพิวเตอร์ในรูปแบบของอาร์เรย์หลายมิติ โดยส่วนที่เล็กที่สุดนั้นเรียกว่า “พิกเซล” (Pixel) อาจกล่าวได้ว่าภาพก็คือพิกเซลที่ประกอบกันเป็นจำนวนมากนั่นเอง และค่าในอาร์เรย์จะอยู่ในรูปของฟังก์ชันสองมิติ f(x,y) โดยที่ x และ y จะแทนพิกัดของจุดภาพ และค่า f เป็นฟังก์ชันแทนความเข้มของจุดภาพนั้น (Intensity)



รูปที่ 2.1 โครงสร้างของภาพดิจิทัล[1]

2.1.2. ข้อมูลเบื้องต้นเกี่ยวกับ Web camera

เว็บแคม (Webcam) คือกล้องวีดีโอที่ใช้ในการส่งภาพแบบเวลาจริง (เวลาจริง) ให้กับคอมพิวเตอร์หรือเครือข่ายคอมพิวเตอร์เครื่องอื่นๆ ประโยชน์ของกล้องเว็บแคมที่นำมาใช้กันมากที่สุดนั่นก็คือ การเชื่อมต่อวีดีโอ ซึ่งเป็นการให้คอมพิวเตอรทำหน้าที่เป็นสถานีในการรับส่งวีดีโอ เช่น สามารถบันทึกภาพนิ่งหรือเคลื่อนไหวของเรา ให้ไปปรากฏในหน้าจอมอนิเตอร์และสามารถส่งภาพเคลื่อนไหวนี้ผ่านระบบเครือข่ายเพื่อให้คนอีกฟากหนึ่งสามารถเห็นตัวเราเคลื่อนไหวได้เหมือนอยู่ต่อหน้า ส่วนประโยชน์ในด้านอื่นที่นำมาใช้ได้แก่ การเฝ้าระวังความปลอดภัย ด้านคอมพิวเตอร์วิทัศน์ (Computer vision) และการแพร่ภาพวีดิทัศน์ (Video broadcasting)

สำหรับเซ็นเซ่อร์ที่กล้องเว็บแคมส่วนมากมีอยู่ 2 ชนิด คือ CCD และ CMOS แต่เซ็นเซ่อร์ที่นิยมใช้กันมากก็คือ CMOS เนื่องจากเหตุผลหลายๆประการและตัวเซ็นเซ่อร์แบบ CMOS ก็สามารถแบบออกได้อีก 2 ชนิดด้วยกัน คือ CLF Color CMOS Censor ที่มีความละเอียดของพิกเซลเพียง 110,000 พิกเซล (367 x 291) ในขณะที่ VGA Color CMOS Censor ให้ความละเอียดที่สูงกว่าที่ 350,000 พิกเซล (655 x 493) สำหรับเซ็นเซอร์แบบ CCD จะเป็นเซ็นเซอร์ที่นิยมใช้ในกล้องดิจิตอล เพราะให้ความละเอียดที่สูงกว่าและมีสัญญาณรบกวนไม่มากเหมือนกับเซ็นเซอร์แบบ CMOS

สำหรับการเชื่อมต่อของกล้องเว็บแคมในปัจจุบันส่วนใหญ่ จะเป็นแบบ USB เกือบทั้งหมด ส่วนกล้องเว็บแคมแบบไร้สายจะใช้การเชื่อมต่อในแบบ Wi-Fi หรือ Wireless LAN ทำให้สามารถเคลื่อนย้ายไปได้ทุกที่ แต่ก็ยังมีระยะที่จำกัดและเว็บแคมที่เชื่อมต่อในแบบ Wireless ค่อนข้างที่จะมีราคาสูงพอสมควร

2.2. ความรู้พื้นฐานด้านคอมพิวเตอร์วิทัศน์

คอมพิวเตอร์วิทัศน์เป็นการศึกษาเกี่ยวกับระบบการคิดวิเคราะห์ และการตัดสินใจเกี่ยวกับรูปภาพ ซึ่งเป็นศาสตร์แขนงหนึ่งทางคอมพิวเตอร์ เพื่อให้คอมพิวเตอร์สามารถเรียนรู้เกี่ยวกับรายละเอียดและข้อมูลที่น่าสนใจในภาพได้ โดนส่วนใหญ่นั้นจะเป็นการศึกษาเกี่ยวกับการตรวจจับและค้นหาวัตถุ การติดตามการเชื่อมโยงของวัตถุ และการประเมินผลเกี่ยวกับสิ่งที่สนใจในภาพ เป็นต้น ซึ่งในปัจจุบันได้มีการศึกษาค้นคว้าและวิจัยเกี่ยวกับศาสตร์แขนงนี้เป็นจำนวนมากและจะกล่าวถึงความรู้พื้นฐานที่เกี่ยวข้องกับด้านคอมพิวเตอร์วิทัศน์ที่ใช้ในปัญหาพิเศษนี้

2.2.1. OpenCV (Open Source Computer Vision Library)

OpenCV คือไลบรารี่ของฟังก์ชันที่ใช้สำหรับการพัฒนางานในด้านการประมวลผลภาพและคอมพิวเตอร์วิทัศน์ที่มีฟังก์ชันมากมายให้เลือกใช้งานตามความเหมาะสม ซึ่ง OpenCV นั้นได้ถูกพัฒนาโดยบริษัทอินเทล (Intel) ด้วย OpenCV เป็นไลบรารีที่ไม่มีข้อจำกัดในเรื่องของฮาร์ดแวร์ (Hardware) จึงสามารถนำใช้งานได้ง่าย และใช้งานได้ในหลายภาษา ไม่ว่าจะเป็นภาษา C ภาษา C++ และภาษา Python เป็นต้น ในไลบรารี่นี้มีชุดคำสั่งที่น่าสนใจ ได้แก่ การค้นหาวัตถุในภาพ การจดจำท่าทางการเคลื่อนไหว การจำแนกวัตถุ และการติดตามการเคลื่อนไหวของวัตถุ เป็นต้น

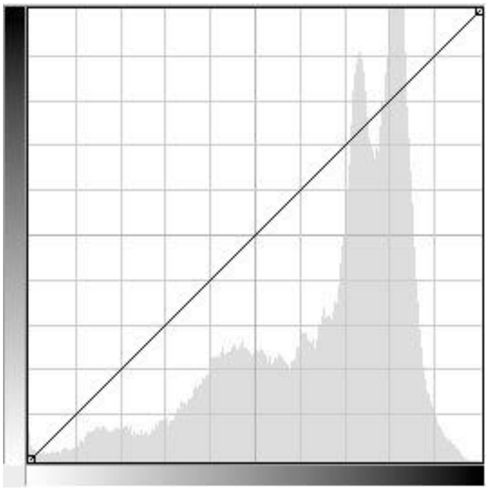
2.2.2. OpenCvSharp

OpenCvSharp คือ OpenCV ที่เปลี่ยนแพลตฟอร์ม (Platform) ไปเป็น .NET Framework โดยใช้ภาษา C# ในการเขียน ซึ่งสามารถใช้การประมวลผลภาพและอัลกอริธึมของคอมพิวเตอร์วิทัศน์โดยใช้ภาษา C# VB .NET หรือภาษาอื่นๆในการพัฒนา อาจกล่าวได้ว่า OpenCvSharp คือ OpenCV ที่ใช้ภาษา C# ในการพัฒนานั่นเอง

2.2.3. Equalization

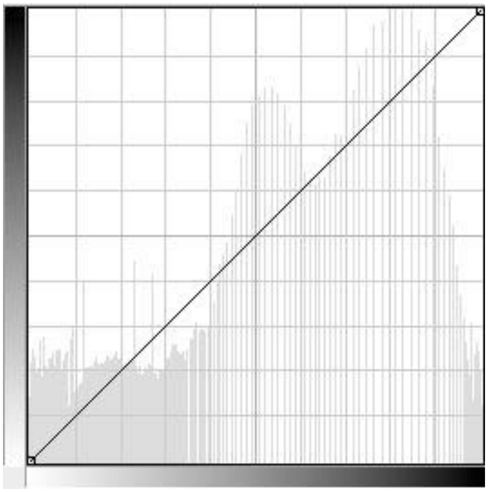
ขั้นตอนการตรวจจับใบหน้า จะมีขั้นตอนการทำงานดังนี้ เริ่มแรก ภาพที่รับเข้ามาเป็นได้ทั้งภาพระดับสีเทาและภาพสีที่ถูกแปลงภาพเป็นภาพระดับสีเทา ภาพสีเทาเหล่านั้นจะถูกนำไปปรับปรุงภาพด้วยวิธี Histogram Equalization เพื่อกระจายให้แต่ละระดับสีเทามีจำนวนพิกเซลใกล้เคียงกัน ซึ่งการทำการกระจายของฮิสโตแกรมนั้น ลักษณะความคมชัดและระดับความสว่างของภาพจะไม่มีผลต่อการแบ่งแยกใบหน้าคน

รูปที่ 2.2 ภาพต้นแบบ



รูปที่ 2.3 กราฟที่สอดคล้องกันของภาพต้นแบบ

รูปที่ 2.4 ภาพหลังจากการทำ Equalization

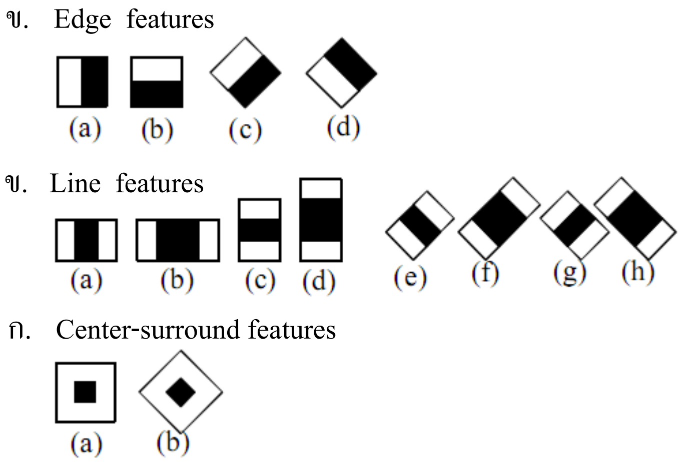


รูปที่ 2.5 กราฟที่สอดคล้องกันของภาพหลังการทำ Equalization

2.2.4. Haar-like Feature

การตรวจหาใบหน้าคนด้วย Haar-like Feature เป็นวิธีทางสถิติ สามารถแบ่งแยกใบหน้าคนได้โดยการเปรียบเทียบลักษณะของ Haar-like Feature จากฐานข้อมูล ที่ผ่านการฝึกให้จดจำลักษณะ Haar-like Feature ของใบหน้าคนและจดจำลักษณะ Haar-like Feature ของสิ่งที่ไม่ใช่ใบหน้าคนไว้แล้ว

นำภาพที่ทำการปรับปรุงแล้วตรวจหาใบหน้าคน โดยทำการเลือกกรอบของพื้นที่ในภาพหาลักษณะ Haar-like Feature เปรียบเทียบกับลักษณะ Haar-like Feature จากฐานข้อมูลทั้งตัวอย่างที่ถูกและตัวอย่างที่ผิด ถ้าลักษณะ Haar-like Featue มีลักษณะของตัวอย่างที่ผิดให้ตัดสินว่าไม่มีใบหน้าคนอยู่ในกรอบที่พิจารณาทันที ทำการเปรียบเทียบกับฐานข้อมูลไปเรื่อยๆจนหมดฐานข้อมูล จึงตัดสินว่ากรอบที่พิจารณาอยู่นั้นมีใบหน้าคนอยู่ กรอบที่ใช้ในการพิจารณานี้เปลี่ยนขนาดในการตรวจหา เพื่อให้ตรวจหาใบหน้าคนได้ในหลายขนาดตามขนาดของกรอบ ขณะเปลี่ยนกรอบตรวจหานี้มีการทำงานพร้อมกันแบบขนาน เพื่อความรวดเร็วในการตรวจหาผลลัพธ์ที่ได้จาก Haar Detection คือตำแหน่งจุดศูนย์กลางของวัตถุที่ถูกพิจารณาว่าเป็นใบหน้าคน และขนาดของกรอบพื้นที่ในการตรวจหา แล้วจึงวงกลมล้อมรอบตำแหน่งที่พบบนภาพที่รับเข้ามาด้วยเส้นผ่านศูนย์กลางที่เท่ากับขนาดของกรอบพื้นที่

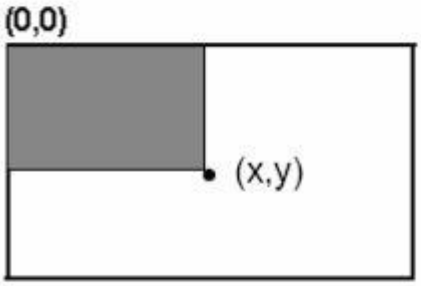


รูปที่ 2.6 รูปแบบของรูปเหลี่ยมสำหรับการตรวจจับลักษณะแบบต่างๆ[2]
(ก) ความสามารถของขอบ (ข) ความสามารถของเส้น
(ค) ความสามารถของบริเวณที่ล้อมรอบจุดตรงกลาง

รูปที่ 2.7 ตัวอย่างการใช้รูปเหลี่ยมตรวจจับลักษณะต่างๆ[3]

การคำนวณค่าของรูปเหลี่ยม (feature) นั้นใช้หลักการคำนวณแบบ Integral image คือผลรวมของค่าในทุกๆจุดภาพที่ตำแหน่ง (x, y) ใดๆดังสมการที่ 2.1

$$P(x,y) = \sum_{x' \le x, y' \le y} i(x',y') \tag{2.1}$$

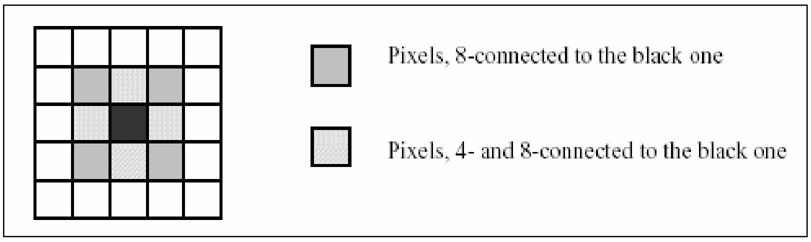


รูปที่ 2.8 การคำนวณแบบ Integral image[2]

ในการทำ Haar like-feature นั้นจำเป็นต้องมีภาพตัวอย่างจำนวนมากซึ่งใช้ในการคัดเลือกลักษณะของรูปที่ต้องการตรวจจับและตีความหมายซึ่งมีสองลักษณะคือ Positive Image หรือรูปที่มีวัตถุนั้นๆประกอบอยู่ภายในภาพและ Negative Image หรือภาพใดๆที่ไม่มีวัตถุที่เราต้องการอยู่ภายในภาพจากนั้นใช้ขั้นตอนวิธีของ AdaBoost (Adaptive Boost) ซึ่งเป็นกระบวนการหารูปเหลี่ยมที่มีลักษณะใกล้เคียงและแตกต่างกับภาพนำเข้าสำหรับการจัดประเภทของภาพโดยการถ่วงน้ำหนักให้ส่วนต่างๆภายในภาพบนภาพ Positive และภาพ Negative เพื่อใช้หาลักษณะของวัตถุที่ “ใช่” และ “ไม่ใช่” ในลักษณะต่างๆ

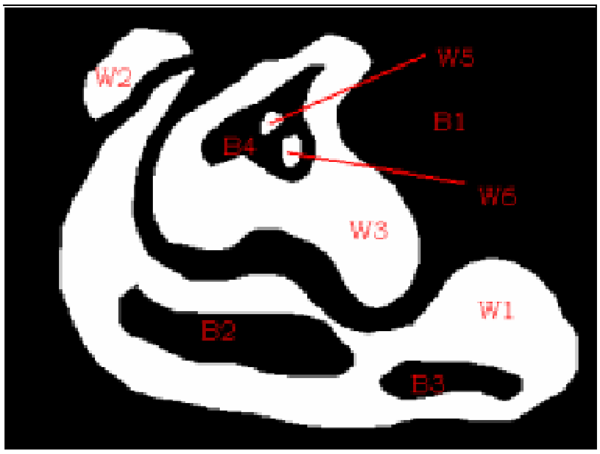
2.2.5. การหาเส้นรูปร่าง

เป็นการใช้ลำดับขั้นตอนของการทำงานเกี่ยวกับเรื่องเวคเตอร์ ซึ่งลำดับขั้นตอนของการทำงานที่ใช้หารูปร่างของแรสเตอร์อิมเมจ (raster image) ในการแปลงเป็นภาพขาวดำ (binary image) ซึ่งภาพขาวดำคือภาพที่มีส่วนประกอบด้วย 0-พิกเซลซึ่งก็คือพิกเซลที่มีค่าเป็น 0 ที่ให้สีพิกเซลเป็นสีดำ และ 1-พิกเซลซึ่งก็คือพิกเซลที่มีค่าเป็น 1 ที่ให้สีพิกเซลเป็นสีขาว เซตของ 0-พิกเซล หรือ 1-พิกเซลที่เชื่อมติดกันจะทำให้เกิด 0-(1-) คอมโพเนนท์ (0-(-1)Component) โดยสามารถแบ่งประเภทของการเชื่อมติดกันเป็น 2 ชนิดได้แก่ แบบ 4-คอนเนคทิวิตี (connectivity) ซึ่งสามารถเขียนให้อยู่ในรูปสมการ $|x'-x''|+|y'-y''|=1$ และ 8-คอนเนคทิวิตี ซึ่งสามารถเขียนให้อยู่ในรูปสมการ โดยกำหนดให้พิกเซล 2 พิกเซลอยู่ที่พิกัด (x', y') และ (x'', y'')



รูปที่ 2.9 รูปแบบการเชื่อมติดกันของพิกเซล[3]

จากความสัมพันธ์ของสมการตามที่กล่าวมานี้รูปจะถูกแบ่งออกเป็นส่วนประกอบเป็นส่วนๆที่ไม่มีการทับซ้อนกัน 1-(0-) 4-คอนเนค (8-คอนเนค) ในแต่ละเซตจะประกอบไปด้วยพิกเซลที่มีค่าที่เท่ากันซึ่งก็คือทั้งหมดเป็นพิกเซลที่มีค่าเป็น 0 หรือ 1 และมีคู่ของพิกเซลที่อยู่ระหว่างจุดสองจุดของอีกเซตรูปส่วนประกอบต่างๆจะมีความสัมพันธ์กันดังรูปที่ 2.10



รูปที่ 2.10 ลำดับชั้นของส่วนประกอบที่เชื่อมติดกัน[3]

1-คอมโพเนนท์ W1, W2 และW3 อยู่ภายในเฟรม (0-คอมโพเนนท์ B1) แสดงว่าถูกล้อมรอบด้วย B1

0-คอมโพเนนท์ B2 และ B3 อยู่ภายใน W1

1-คอมโพเนนท์ W5 และ W6 อยู่ภายใน B4 ซึ่งอยู่ภายใน W3 ดังนั้น 1-คอมโพเนนท์อยู่ภายใน W3 โดยตรงแต่ทั้ง W5 และ W6 ไม่ได้อยู่ติดกับส่วนอื่น หมายความว่าส่วนที่กล่าวถึงนี้อยู่คนละระดับชั้นกันกับส่วนอื่น

หากต้องการหลีกเลี่ยงความขัดแย้งที่เกิดจากโครงสร้าง 0-พิกเซลจะถือว่าเป็นพิกเซล 8-(4-) คอนเนคในกรณีที่ 1-พิกเซลจะเกี่ยวข้องกับ 4-(8-) คอนเนคโดยจะสมมติว่าจากภาพนี้ 1-พิกเซลเป็น 8-คอนเนคและ 0-พิกเซลเป็น 4-คอนเนค

จากการที่ 0-คอมโพเนนท์จะเสริมให้ 1-คอมโพเนนท์มีความสมบูรณ์และแยก 1-คอมโพเนนท์ให้ออกจากชิ้นส่วนอื่นที่ไม่ติดกัน ไลบรารี่ OpenCV จึงพิจารณาเพียงแค่ส่วนที่เป็นโครงสร้างในการศึกษาเท่านั้นส่วนที่เป็น 0-พิกเซลจะถือว่าเป็นภาพพื้นหลังในส่วนของ 0-คอมโพเนนท์ที่ถูกล้อมรอบด้วย 1-คอมโพเนนท์โดยตรงก็จะเรียกว่าหลุมโฮลล์ (hole) ของ 1-คอมโพเนนท์จุดในเส้นขอบของ 1-คอมโพเนนท์ก็คือพิกเซลใดๆที่อยู่ในส่วนนั้นและมี 4-คอนเนคเชื่อมกับ 0-พิกเซลเซตของจุดในเส้นขอบนั้นจะเรียกว่าเส้นขอบ (border)

ในแต่ละ 1-คอมโพเนนท์จะมีเส้นขอบภายนอก (outer border) ซึ่งจะแบ่งตัวเองออกจาก 0-คอมโพเนนท์ที่ล้อมรอบอยู่และอาจจะมีเส้นขอบของหลุมโฮลล์ (hole border) มากกว่า 1 หลุมโฮลล์ที่แบ่งส่วนของ 1-คอมโพเนนท์ออกจาก 0-คอมโพเนนท์ที่อยู่รอบๆจะเห็นได้ชัดว่าเส้นขอบภายนอกและเส้นขอบหลุมโฮลล์จะทำให้สามารถอธิบายรูปของส่วนประกอบนั้นได้จนครบ

ดังนั้นเมื่อมีเส้นขอบทั้งหมดก็สามารถชี้ให้เห็นถึงเส้นรูปร่าง (contour) ของทุกๆชิ้นส่วนได้ด้วยการเก็บข้อมูลทั้งหมดเกี่ยวกับการจัดลำดับขั้นแทนด้วยการบีบอัดภาพของรูปไบนารีต้นแบบ

2.2.6. บริเวณที่สนใจ Region-of-interest (ROI)

Region-of-interest (ROI) คือบริเวณที่เราสนใจ โดยการตีกรอบล้อมรอบบริเวณที่สนใจด้วยวงกลม กรอบสี่เหลี่ยม หรือกรอบรูปเหลี่ยมใดๆ ซึ่งอาจจะเป็นบริเวณใดภายในภาพก็ได้ เพื่อนำภาพเฉพาะส่วนดังกล่าวมาประมวลผล หรือเปลี่ยนแปลงภาพตามต้องการ โดยไม่มีผลกระทบกับส่วนอื่นๆ ซึ่งภายในหนึ่งภาพสามารถกำหนดได้หลายๆบริเวณที่สนใจ

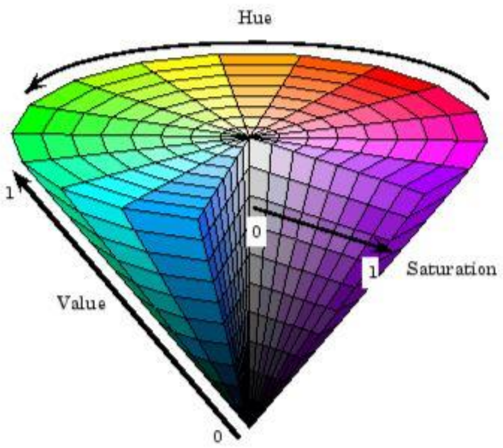
2.3. ระบบสี

2.3.1. ความรู้พื้นฐานเกี่ยวกับภาพสีระบบอาร์จีบี (RGB Image)

ภาพสีระบบอาร์จีบีเป็นที่นิยมอย่างมากในการบันทึกภาพดิจิทัล โดยภาพจะแสดงสีในแต่ละพิกัดออกมาจากการผสมกันของ สีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) ภาพสีระบบอาร์จีบีนั้นจะถูกเก็บอยู่ในรูปแบบของอาร์เรย์ 3 มิติที่มีขนาด m x n x 3 โดยที่ m และ n แทนขนาดความยาวและความกว้างของภาพ ส่วนมิติสุดท้ายนั้นจะท าการเก็บค่าของสีแดง สีเขียว และสีน้ำเงิน

2.3.2. ความรู้พื้นฐานเกี่ยวกับภาพสีระบบ HSV

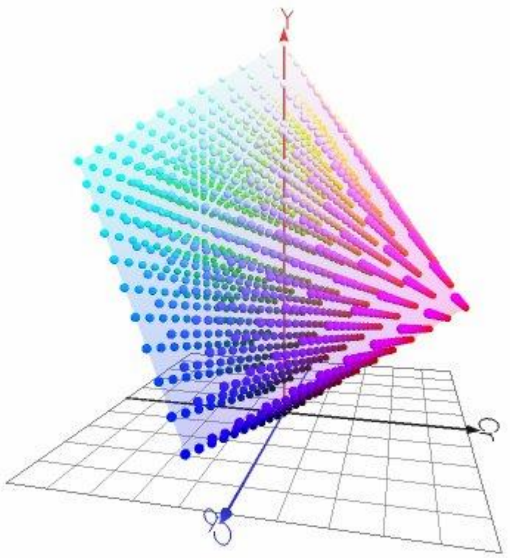
ประกอบด้วย 3 องค์ประกอบได้แก่ ค่า H (Hue) คือค่าโทนสี หรือค่าสีสัน ซึ่งมีค่าสีแตกต่างกันไปตามความถี่ของแสง ค่า S (Saturation) คือค่าความอิ่มตัวสี ซึ่งเป็นค่าที่แสดงระดับสีเมื่อเทียบกับค่าโทนสี และค่า V (Value)หรือB (Brightness)คือค่าบอกระดับความสว่างของภาพ ซึ่งที่ระดับความสว่างต่ำสุดหมายถึงสีดำ ไม่ว่าค่าโทนสีหรือค่าความอิ่มตัวสีเท่าใด และระดับความสว่างสูงสุดหมายถึงสีขาว ซึ่งเป็นสีที่สว่างที่สุดของค่าโทนสีและค่าความอิ่มตัวสี



รูปที่ 2.11 แบบจำลองสี HSV[4]

2.3.3. ความรู้พื้นฐานเกี่ยวกับภาพสีระบบ YCbCr

YCbCr , Y′CbCr, or Y Pb/Cb Pr/Cr, หรือสามารถเขียนเป็น YC_BC_R or $Y'C_BC_R$ เป็นหนึ่งในระบบการเข้ารหัสสัญญาณสีที่นำมาใช้ในวีดีโอ Component และระบบระบบภาพดิจิตอล ความแตกต่างระหว่าง YCbCr กับ RGB ก็คือ YCbCr แสดงภาพโดยใช้สัญญาณความสว่างและสัญญาณความต่างสี (Color Difference) อีกสองสัญญาณ ส่วน RGB แสดงภาพด้วยสัญญาณสีแดง เขียว และน้ำเงิน คำว่า YCbCr ตัวอักษร Y มาจากความสว่าง (Luminance) Cb คือสีน้ำเงินที่ตัดความสว่างออกไป (B-Y) Cr คือสีแดงที่ตัดความสว่างออกไป (R-Y)



รูปที่ 2.12 แบบจำลองสี YCbCr[5]

2.4. งานวิจัยที่เกี่ยวข้อง

จากการศึกษาของ Rahman et al.[6] เพื่อกำหนดขอบเขตพื้นที่สีผิวจากการวิเคราะห์การกระจายค่าข้อมูลสีผิว โดยใช้โมเดลสีทั้งสามได้แก่ RGB, HSV และ YCbCr ที่ใช้ในงานวิจัยนี้เกี่ยวกับการการตรวจจับพื้นที่ใบหน้าโดยใช้โมเดลสี RGB-H-CbCr ในโมเดล

กฎการกำหนดขอบเขตพื้นที่สีผิวจากโมเดลสี RGB นำเสนอโดย Peer et al.[7] ซึ่งกฎการกำหนดของเขตพื้นที่สีผิวโดยโมเดลสี RGB แบ่งเป็นบริเวณพื้นที่สีผิวที่ได้รับแสงในช่วงกลางวันและบริเวณพื้นที่สีผิวที่ได้รับแสงในช่วงกลางคืน ซึ่งบริเวณพื้นที่สีผิวที่ได้รับแสงในช่วงกลางวันสามารถหาได้จากสมการดังนี้

$$\begin{gathered}\mathbf{(R > 95)AND(G > 40)AND(B > 20)\ \ AND} \\ \mathbf{(max\{R, G, B\} - min\{R, G, B\} > 15)\ \ AND} \\ \mathbf{(|R - G| > 15)AND(R > G)AND(R > B)}\end{gathered} \tag{2.2}$$

ในขณะบริเวณพื้นที่สีผิวที่ได้รับแสงในช่วงกลางคืน จะหาได้จากสมการดังนี้

$$\begin{gathered}\mathbf{(R > 220)AND(G > 210)AND(B > 170)\ \ AND} \\ \mathbf{(|R - G| \le 15)AND(R > B)AND(G > B)}\end{gathered} \tag{2.3}$$

เพื่อพิจารณาเงื่อนไข จะใช้ลอจิก OR ในการรวมสมการทั้งสองเข้าด้วยกันคือ สมการที่ 2.2 OR กับสมการที่ 2.3 จะได้กฎการกำหนดขอบเขตพื้นที่สีผิวโดยโมเดลสี RGB

$$\textbf{สมการที่}(\mathbf{2.2})\ \cup\ \textbf{สมการที่}(\mathbf{2.3}) \tag{2.4}$$

สำหรับนำไปใช้ในการตรวจจับพื้นที่บริเวณสีผิว เพื่อกำหนดขอบเขตบริเวณพื้นที่สีผิวโดยวิเคราะห์การกระจายค่าข้อมูลสีผิวในโมเดลสี YCbCr พบว่าการกระจายค่าข้อมูลสีผิวในค่าข้อมูลสี Cb และ Cr จะเกาะกลุ่มอยู่ในช่วงตรงกลางระหว่าง 0 ถึง 255 จึงทำให้สามารถทราบค่าข้อมูลสีผิวที่ชัดเจน ทฤษฎีนี้จึงทำการประมาณการค่าข้อมูลโดยนำคณิตศาสตร์พื้นฐานสมการเชิงเส้นมาใช้ในการหาขอบเขตพื้นที่การกระจายค่าข้อมูลสีผิว จะได้สมการดังนี้

$$\mathbf{Cr \le 1.5862 \times Cb + 20} \tag{2.5}$$

$$\mathbf{Cr \ge 0.3448 \times Cb + 76.2069} \tag{2.6}$$

$$\mathbf{Cr \ge -4.5652 \times Cb + 234.5652} \tag{2.7}$$

$$\mathbf{Cr \le -1.15 \times Cb + 301.75} \tag{2.8}$$

$$\mathbf{Cr} \leq -\mathbf{2.2857} \times \mathbf{Cb} + \mathbf{432.85} \tag{2.9}$$

เพื่อพิจารณาเงื่อนไข จะใช้ลอจิก AND ในการรวมสมการที่ 2.5-2.9 เข้าด้วยกันคือ สมการที่ 2.5 AND กับสมการที่ 2.6 AND กับสมการที่ 2.7 AND กับสมการที่ 2.8 AND กับสมการที่ 2.9 จะได้กฎการกำหนดขอบเขตพื้นที่สีผิวโดยโมเดลสี YCbCr

$$\text{สมการที่}(2.5) \cap \text{สมการที่}(2.6) \cap \text{สมการที่}(2.7) \cap \text{สมการที่}(2.8) \cap \text{สมการที่}(2.9) \tag{2.10}$$

โมเดลสี HSV จากการวิเคราะห์การกระจายค่าข้อมูลสีผิวในโมเดลสี HSV พบว่าการกระจายค่าข้อมูลสีผิวในค่าสี (Hue) แสดงให้เห็นได้ชัดเจนที่สุดในการแบ่งแยกระหว่างสีผิวและไม่ใช่สีผิว ซึ่งดูได้จากความเข้มข้นของค่าสี (Hue) จะได้สมการดังนี้

$$(\mathbf{H} < \mathbf{5})\mathbf{OR}(\mathbf{H} > \mathbf{130}) \tag{2.11}$$

กฎการกำหนดขอบเขตพื้นที่สีผิวโดยโมเดลสี HSV

2.5. Unity

Unity คือเกมเอนจิ้นที่ถูกพัฒนาจากเทคโนโลยีของ Unity สำหรับนักพัฒนาทั่วโลกกว่าหนึ่งล้านคน เพื่อให้สามารถรองรับได้หลากหลายแพลตฟอร์มและมีเครื่องมือในการพัฒนาโปรแกรมเพื่ออำนวยความสะดวกต่างๆในการพัฒนาวีดีโอเกมที่ใช้สำหรับเล่นบนเว็บไซต์ แพลตฟอร์มของคอมพิวเตอร์ เครื่องมือควบคุมวีดีโอเกม (เช่น joystick) และอุปกรณ์มือถือ ในช่วงแรกของ Unity เป็นเครื่องมือในการพัฒนาเกมที่รองรับเพียงเฉพาะแพลตฟอร์มโอเอสเอกซ์ หลังจากนั้นจึงมีการพัฒนาเรื่อยมา จนกระทั่งสามารถรองรับได้หลายแพลตฟอร์มในปัจจุบัน และโดยใช้ภาษา C# หรือ C++ ในการพัฒนา มีการปรับปรุงรุ่นล่าสุดคือ Unity4.2 ซึ่งถูกปล่อยออกมาเมื่อเดือนกรกฎาคม ปีค.ศ.2013 โดยสามารถรองรับการพัฒนาจากทั้ง ไอโอเอส แอนดรอยด์ วินโดส์ แบล็กเบอร์รี่ 10 โอเอสเอกซ์ ลินุกซ์ เว็บบราว์เซอร์ แฟลช เพลย์สเตชัน 3 เอกซ์บอกซ์ 360 วินโดส์โฟน 8 และ วียู ซึ่งเปิดให้สามารถดาวน์โหลดแล้วทั้ง Unity และ Unity Pro

2.5.1. การแสดงผล

กราฟฟิคเอนจิ้นที่ใช้ได้แก่ Direct3D (วินโดส์ เอกซ์บอกซ์ 360) OpenGL (แมค วินโดส์ ลินุกซ์ PS3) OpenGL ES (แอนดรอยด์ ไอโอเอส) และ proprietary APIs (API ที่ถูกทำให้ใช้ได้เฉพาะกับอุปกรณ์ชนิดใดชนิดหนึ่งเท่านั้น(Wii)) โดยสามารถรองรับการทำความขรุขระของพื้นผิว การทำให้วัตถุมีความสามารถในการสะท้อน การทำให้พื้นผิวมีความลึกและสมจริง การเพิ่มความสมจริงของเงาจากแสงสะท้อนรอบทิศทาง การคำนวนเงาโดยใช้กระบวนการใส่เงาลงไปในงานคอมพิวเตอร์กราฟฟิคสามมิติ การนำแสงที่ได้จากการคำนวณจากการ render มาใส่ลงใน texture และ full-screen post-processing effects

Unity สามารถรองรับไฟล์งานจากโปรแกรม 3ds Max, Maya, Softimage, Blender, modo, Zbrush, Cinema 4D, Adobe Photoshop, Adobe Fireworksและ Allegorithmic Substance ซึ่งสามารถเพิ่มเข้าไปในโปรเจคเกมและจัดการไฟล์เหล่านั้นผ่านหน้าต่างการใช้งานกราฟฟิคของ Unity

ภาษาที่ใช้ในการทำ shaders คือภาษา ShaderLab สามารถรองรับทั้งหลักการเขียนโปรแกรมแบบfixed-function pipeline และ shader ที่ใช้เขียนใน GLSL(OpenGL Shading Language) หรือ Cg(C for Graphics) โดยทั้ง GLSL และ Cg ต่างก็เป็นภาษา shading ระดับสูงเช่นเดียวกัน ซึ่งภายใน shader สามารถมีตัวแปรที่หลากหลายและสามารถประกาศการย้อนกลับในกรณีเฉพาะ และยังอนุญาติให้ Unity ใช้ตรวจจับตัวแปรให้เหมาะสมกับการ์ดจอ ซึ่งถ้าหากไม่มีการ์ดจอที่เข้ากันได้ ก็จะทำการย้อนกลับเพื่อหาทางเลือกใหม่ของ shader ซึ่งอาจจะเสียคุณลักษณะบางประการในการแสดงผล

Unity 4.2 รุ่นล่าสุดที่ปล่อยออกมาเมื่อวันที่ 3 สิงหาคม 2013 นั้นได้อนุญาติให้นักพัฒนาอิสระสามารถใช้การให้เงาแบบ เวลาจริง เฉพาะสำหรับการให้แสงแบบมีทิศทางเท่านั้น และเพิ่มการรองรับ DirectX11 ที่ให้เงาที่มีความละเอียดของพิกเซลที่สมบูรณ์มากขึ้นเพิ่มพื้นผิวในการสร้างวัตถุสามมิติจากภาพระดับสีเทา กราฟฟิคใบหน้าที่มากขึ้น การทำอนิเมชันที่เรียบเนียนกว่าเดิม และรองรับการทำให้เป็น FPS

2.5.2. การเขียนสคริปต์

สคริปต์ของเกมเอนจิ้นถูกสร้างขึ้นจาก Mono2.6 ซึ่งเป็นโอเพนซอร์ซที่ใช้ .NET Framework โดยโปรแกรมเมอร์สามารถใช้สคริปต์ Unity (ภาษาที่กำหนดขึ้นเองและใช้โครงสร้าง

จากสคริปต์ ECMAโดยอ้างอิงกับ JavaScript) C# หรือ Boo (ใช้โครงสร้างจากภาษา Python) โดยตั้งแต่รุ่น 3.0 เป็นต้นไปการปรับปรุงสคริปต์เพื่อแก้ไขจุดบกพร่องจากทีมพัฒนาของบริษัทโมโนจะถูกรวมเข้ากับ Unity ไว้ด้วยกัน

2.5.3. Asset Tracking

Unity มีการรวม Unity Asset Server ซึ่งเป็นวิธีการควบคุมรุ่นสำหรับผลงานและสคริปต์ของนักพัฒนาเกม มีการใช้ PostgreSQL เป็นส่วนในการจัดการระบบ ระบบของเสียงนั้นสร้างจากไลบรารี่ FMOD(ที่มีความสามารถในการเล่นเสียงที่ผ่านการบีบอัดมาแล้วของบริษัท Ogg Vorbis) ใช้ Theora codec ในการเล่นวีดีโอ มีเอนจิ้นการทำภูมิศาสตร์และพฤกศาสตร์(ใช้ Umbra ที่รองรับ tree billboarding และ Occlusion Culling)มี light mapping และการให้แสงแบบทั้งระบบ จาก Beast มีการใช้ RakNet ในการติดต่อกับผู้เล่นจำนวนหลายคน และมีการค้นหาเส้นทางแบบตาข่าย

2.5.4. แพลตฟอร์ม

สามารถรองรับการใช้งานหลากหลายแพลตฟอร์ม นักพัฒนาสามารถควบคุมโปรเจคที่จะพัฒนาสำหรับอุปกรณ์มือถือ เว็บบราว์เซอร์คอมพิวเตอร์และเครื่องมือควบคุมวีดีโอเกม(เช่น joystick) โดยสามารถตั้งค่าเฉพาะสำหรับการแปลงพื้นผิวและความละเอียดของเกมในแต่แพลตฟอร์มให้มีความแตกต่างกันออกไปได้อีกด้วย

ในปัจจุบันสามารถรองรับแพลตฟอร์ม ไอโอเอส แอนดรอยด์ วินโดส์ แมค ลินุกซ์ Unity web Player Adobe Flash เพลย์สเตชัน 3 เอกซ์บอกซ์ 360 และ Wii ซึ่งยังสามารถรองรับเพลย์สเตชัน Vita ที่สามารถพบเห็นได้ในเกม Escape Plan และ Oddworld:New'n'Tasty แต่ข่าวนี้ยังไม่ได้รับการยืนยันอย่างเป็นทางการจากทาง Unity

Unity รุ่นล่าสุดที่กำลังมาสามารถรองรับแพลตฟอร์ม แบล็กเบอร์รี่ 10 เพลย์สเตชัน 4 วียูเอกซ์บอกซ์วัน วินโดส์โฟน 8 ทั้งยังมีข่าวลืออีกว่า ในอนาคตจะสามารถรองรับ HTML และ Adobe plug-in รุ่นใหม่ที่จะมาแทนที่ Flash Player อีกด้วย

2.5.5. Asset Store

Unity Asset Store เปิดตัวครั้งแรกเมื่อเดือนพฤศจิกายนปี 2010 ซึ่งเป็นแหล่งทรัพยากรที่อยู่ภายใน Unity editor โดยมีการรวบรวมผลงานเป็นจำนวนมากกว่า 4,400 ชิ้น ประกอบไปด้วยโมเดลสามมิติ พื้นผิวและวัสดุ ชิ้นส่วนของระบบ เพลงและเสียงเอฟเฟค คู่มือและโปรเจคชุดคำสั่งสคริปต์ส่วนแก้ไขเพิ่มเติมและการบริการออนไลน์

2.5.6. ฟิสิกส์

Unity มีการรองรับ Nvidia's(ชื่อเรียกอย่างเป็นทางการคือ Ageia's) PhysX ซึ่งเป็นฟิสิกส์เอนจิ้น(นำมาใช้ตั้งแต่Unity 3.0) ที่รองรับการจำลองการเคลื่อนไหวของเสื้อผ้าและผิวหนังการคำนวนความทึบโปร่ง และการชนกันแบบ เวลาจริง

2.5.7. รุ่น

Unity รุ่นแรกได้เปิดตัวในงานประชุมนักพัฒนา Apple จากทั่วโลกในปี 2005 ซึ่งฟังก์ชันการทำงานสร้างมาเพื่อรองรับโปรเจคที่ทำงานบนคอมพิวเตอร์แมคเท่านั้นและเริ่มพัฒนาให้สามารถรองรับแพลตฟอร์มอื่นๆเรื่อยมา โดย Unity 3 ที่ถูกปล่อยออกมาในเดือนกันยายนปี 2010 ได้มุ่งเน้นไปที่การแนะนำเครื่องมือที่สตูดิโอระดับใหญ่สามารถนำไปใช้ได้ ซึ่งเป็นการทำให้บริษัทหันเหความสนใจไปที่กลุ่มนักพัฒนาที่มีขนาดใหญ่ขึ้น รวมไปถึงการทำเกมเอนจิ้นแพคเกจสำหรับนักพัฒนาอิสระและทีมพัฒนาขนาดเล็กให้สามารถซื้อหาได้ โดยรุ่น 4.0 ที่ถูกปล่อยออกมาล่าสุดในปลายปี 2012 ได้มีการเพิ่มการรองรับ Mecanium animation และ DirectX11

2.5.7.1. Unity 4

Unity 4 มีการประกาศในวันที่ 18 มิถุนายน 2012 ว่าจะมีการเพิ่มเทคโนโลยีเล็กๆน้อยๆดังจะคล้ายกับการปล่อยออกมาของรุ่นก่อนๆ ซึ่งมีทั้งการปรับปรุงและเพิ่มคุณสมบัติบางอย่างเช่นการปรับปรุงGUIใหม่ และมีการจำหน่ายจริงในวันที่ 14 พฤศจิกายน 2012

คุณสมบัติใหม่ที่เพิ่มขึ้นมานั้นได้แก่ การรองรับ DirectX 11 และ Mecanim animation ซึ่งภายในกราฟฟิคของอุปกรณ์สื่อสารมีการให้เงาแบบเวลาจริง skinned mesh instancing, ความสามารถในการใช้แผนที่ธรรมดาเมื่อมีการใช้ baking lightmaps และการคัดกรอง GPU profiler อย่างไรก็ตามการใช้งาน Adobe Flash add-on สามารถใช้กับ Unity รุ่น 4.0 ขึ้นไปเท่านั้น (รองรับเฉพาะกับเครื่องมือของ Unity 3.5 ที่เป็นรุ่น beta) และยังมีความลือว่า Unity 4.2 จะเพิ่มคุณสมบัติของเงาที่รองรับการทำงานสำหรับรุ่นฟรี แต่ก็ยังมีข้อจำกัดบางประการเช่น อาจจะจำกัดให้สามารถใช้แสงได้เพียงทิศทางเดียว เป็นต้น

Unity 4 เป็นการเพิ่มทางเลือกในการทำให้เกมสามารถเล่นได้หลากหลายแพลตฟอร์มและมีการเริ่มต้นการพัฒนาไปสู่การใช้งานบน Ubuntu โดยไม่เสียค่าใช้จ่ายเพิ่มเติมใดๆทั้งสิ้นและตั้งแต่ Unity 4.0 เป็นต้นไป จะมีการร่วมงานกับทาง Facebook เพื่อรองรับสำหรับแพลตฟอร์มโซเชียลซึ่งมีชื่อเรียกว่า Unity Web Player

2.5.7.2. Mecanim

Mecanim คือเทคโนโลยีแอนิเมชันของ Unity ที่มีการพัฒนามาหลายปีแล้วจากบริษัทเล็กๆที่มีชื่อเดียวกันนี้ ซึ่งภายหลังทาง Unity สาขาประเทศแคนาดาได้เข้ามาซื้อกิจการ โดยเทคโนโลยีนี้เป็นการนำเอาสสารและการเคลื่อนไหวของธรรมชาติมาสร้างลักษณะ ซึ่งมีเครื่องมือสำหรับการสร้าง state machines, blend tree, IK rigging และแอนิเมชันการกำหนดเป้าหมายใหม่โดยอัตโนมัติที่อยู่ภายใน Unity editor