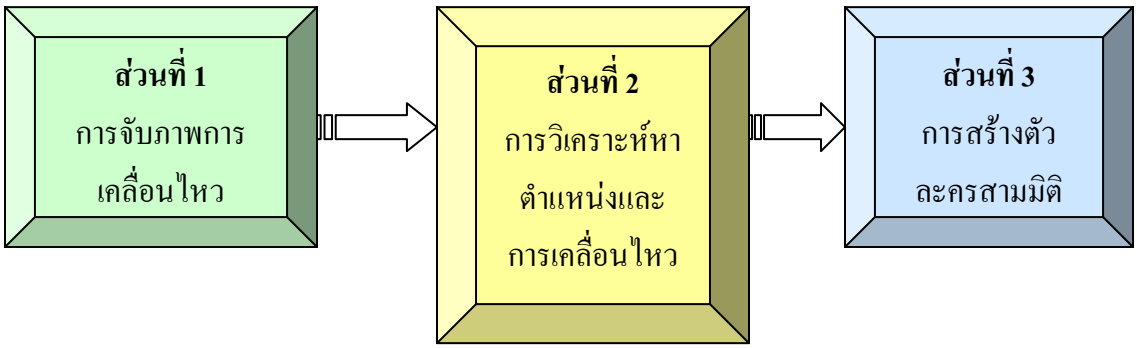
บทที่ 3

การออกแบบระบบ

3.1 ระบบโดยรวม

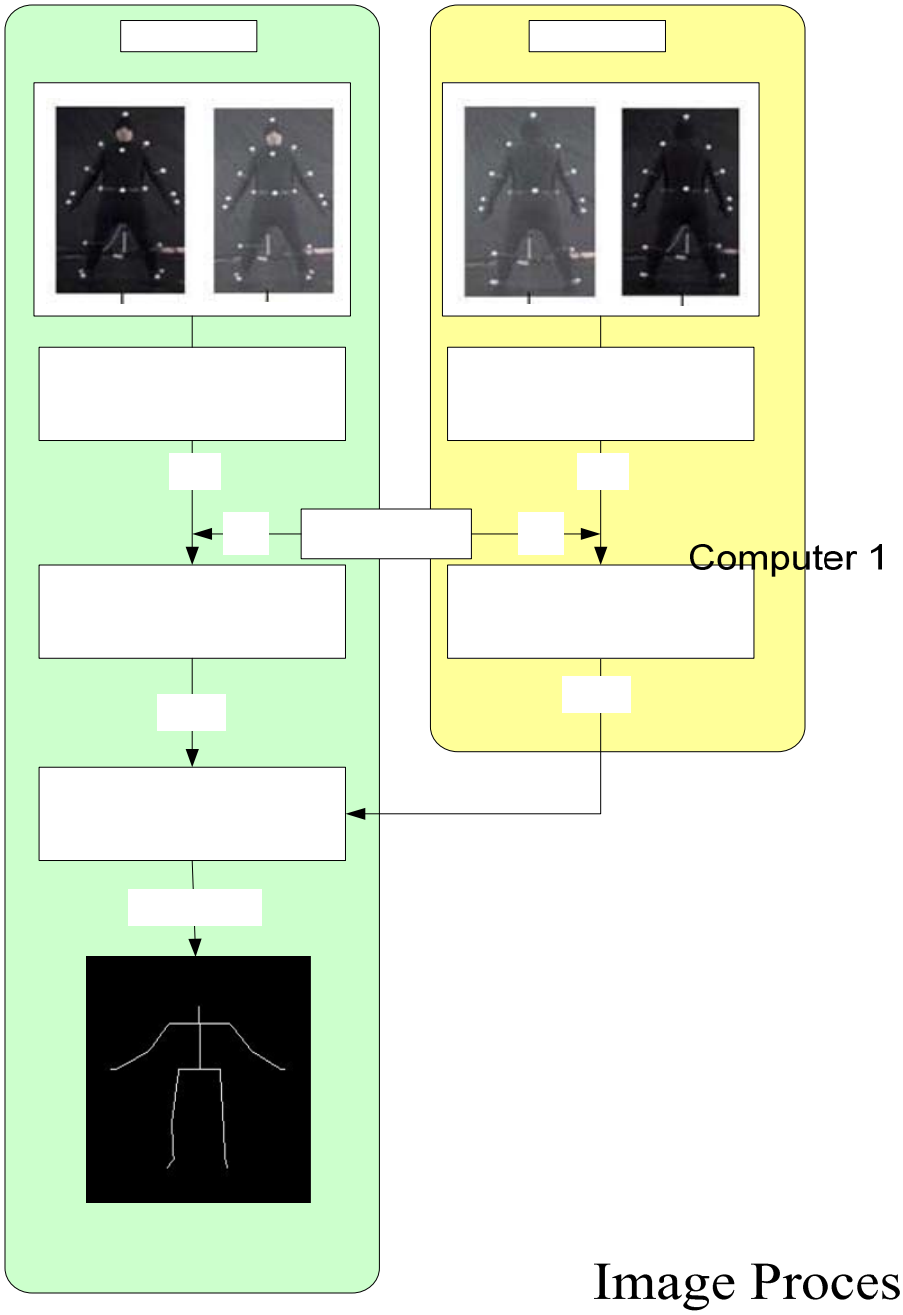


รูปที่ 3.1 ระบบโดยรวม

1. การบันทึกภาพการเคลื่อนไหว (Motion Record) เป็นส่วนที่ต้องจัดเตรียมทุกอย่างก่อนการทำงาน ได้แก่อุปกรณ์ต่างๆ การตั้งกล้อง การจัดสถานที่ การจัดแสง ชุดที่ใช้ใส่แสดง และผู้แสดง (Actor) ระบบประมวลผล เพื่อทำการจับภาพวีดีโอจากกล้องทั้ง 4 ตัวแล้วส่งภาพแต่ละเฟรมไปส่วนต่อไป
2. การวิเคราะห์หาการเคลื่อนไหว (Motion Analysis) ใช้การประมวลผลภาพเพื่อหาตำแหน่ง
3. มาร์กเกอร์แต่ละจุดของแต่ละเฟรมของภาพที่ได้จากข้อ 1 แล้วนำมาคำนวณหาตำแหน่งจริง (World Coordinate) แล้วส่งข้อมูลตำแหน่งการเคลื่อนไหวที่คำนวณได้ไปให้ส่วนต่อไป
4. การสร้างการเคลื่อนไหวให้กับตัวละคร (Animation Production) เป็นขั้นตอนที่จะนำเอาข้อมูลตำแหน่งการเคลื่อนไหว (Motion Data) ไปกำหนดให้กับตัวละคร 3 มิติ ในที่นี้ใช้ OpenGL เพื่อสร้างโครงสร้างโครงกระดูกมนุษย์แล้วจึงส่งข้อมูลตำแหน่งเหล่านั้นให้ตัวโครงกระดูกแสดงผลการเคลื่อนไหวตามข้อมูลที่ได้มานั้น

3.2 การออกแบบโปรแกรม

การจับภาพจากกล้องวิดีโอทั้ง 4 ตัวพร้อมๆ กันนั้น จำเป็นที่จะต้องใช้การ์ดจับภาพ (Video Capture Card) 4 ชุด ซึ่งมีข้อจำกัดตรงที่คอมพิวเตอร์ 1 เครื่อง สามารถติดตั้งการ์ดได้เพียง 2 ชุด ดังนั้นจึงต้องใช้คอมพิวเตอร์ 2 เครื่องในการจับภาพและส่งผลมาทำการแสดงในเครื่องใดเครื่องหนึ่ง



รูปที่ 3.2 การออกแบบโปรแกรม

(x,y)s

Initial value

Calibra

Image Processing

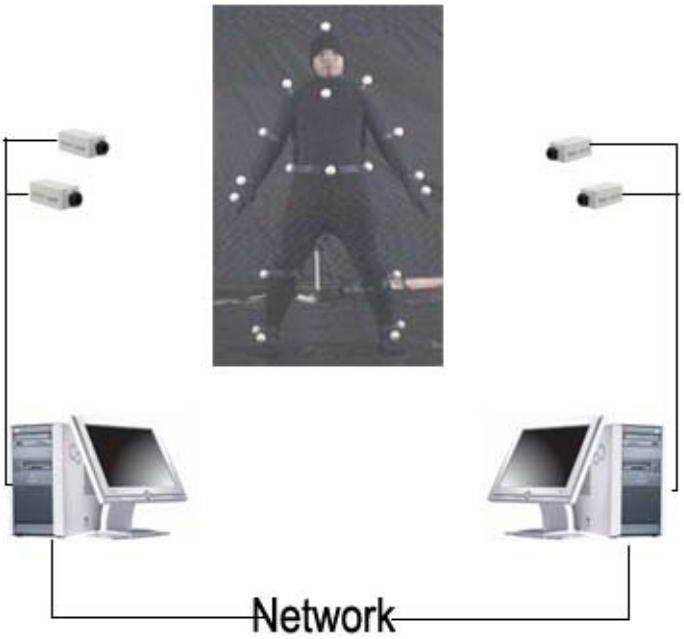
3.2.1 ขั้นตอนการทำงานของโปรแกรม

1. เครื่องคอมพิวเตอร์ทั้งสองเครื่องทำการจับภาพจากกล้อง 4 ตัวเครื่องละสองกล้อง
2. นำภาพที่ได้มาจากแต่ละกล้องส่งต่อให้ส่วน Image Processing ทำการหาตำแหน่ง x,y ในหนึ่งเครื่องจะสามารถคำนวณภาพจากกล้องสองตัว
3. หลังจากได้รับตำแหน่ง x และ y ของมาร์กเกอร์แต่ละตัวแล้วก็นำตำแหน่งแรกที่เจอส่งไปให้กับส่วน Calibration ทำการหาค่าตัวแปรให้กับฟังก์ชัน Stereopsis
4. จากนั้นก็นำตำแหน่งที่เจอทั้งหมดแรกที่เจอส่งไปให้กับฟังก์ชัน Stereopsis เพื่อหาตำแหน่ง 3 มิติ x, y และ z
5. หลังจากผ่านฟังก์ชัน Stereopsis เครื่องแต่ละเครื่องก็จะได้ตำแหน่ง 3 มิติเครื่องละชุด
6. ทำการ Initial post เพื่อให้ได้ค่าค่าเริ่มต้นไปใช้ในฟังก์ชัน Tracking
7. ส่งผลลัพธ์ตำแหน่ง 3 มิติแต่ละชุดไปให้ฟังก์ชัน Tracking เพื่อระบุตำแหน่ง
8. เมื่อรู้ว่าตำแหน่งแต่ละอันเป็นตำแหน่งไหนก็ส่งไปให้โครงกระดูกแสดงผล
9. จากนั้นก็กลับไปเริ่มขั้นตอนรับภาพมาคำนวณใหม่

3.2.2 จากการทำงานของระบบทั้งหมดสามารถแบ่งการทำงานโปรแกรมได้ 7 ส่วน

1. ส่วนการรับภาพ ต้องเขียนโปรแกรมขึ้นมาติดต่อกับ Capture Cards เพื่อรับภาพ
2. ส่วน User Interface เขียนโปรแกรมขึ้นมาเพื่อเป็นตัวกลางให้ผู้ใช้ระบบสามารถปรับค่าที่จำเป็นต่างๆ
3. ส่วน Calibration หาค่าต่างๆ ที่จำเป็นต้องใช้ในส่วน Stereopsis ส่วนนี้ใช้ Tool ที่มีอยู่ในโปรแกรม Matlab ในการทำงาน
4. ส่วน Stereopsis เขียนโปรแกรมขึ้นมาหาตำแหน่งจริง (World Coordinate)
5. ส่วน Initial เขียนโปรแกรมให้ระบบรู้จักมาร์กเกอร์ว่ามาร์กเกอร์ที่พอเป็นมาร์กเกอร์ส่วนไหน
6. ส่วน Tracking เขียนโปรแกรมเพื่อติดตามการเคลื่อนไหวของมาร์กเกอร์
7. ส่วน Render แสดงโครงสร้างโครงกระดูกจากผลที่ระบบหามาได้

3.3 การจับภาพการเคลื่อนไหว (Motion Record)



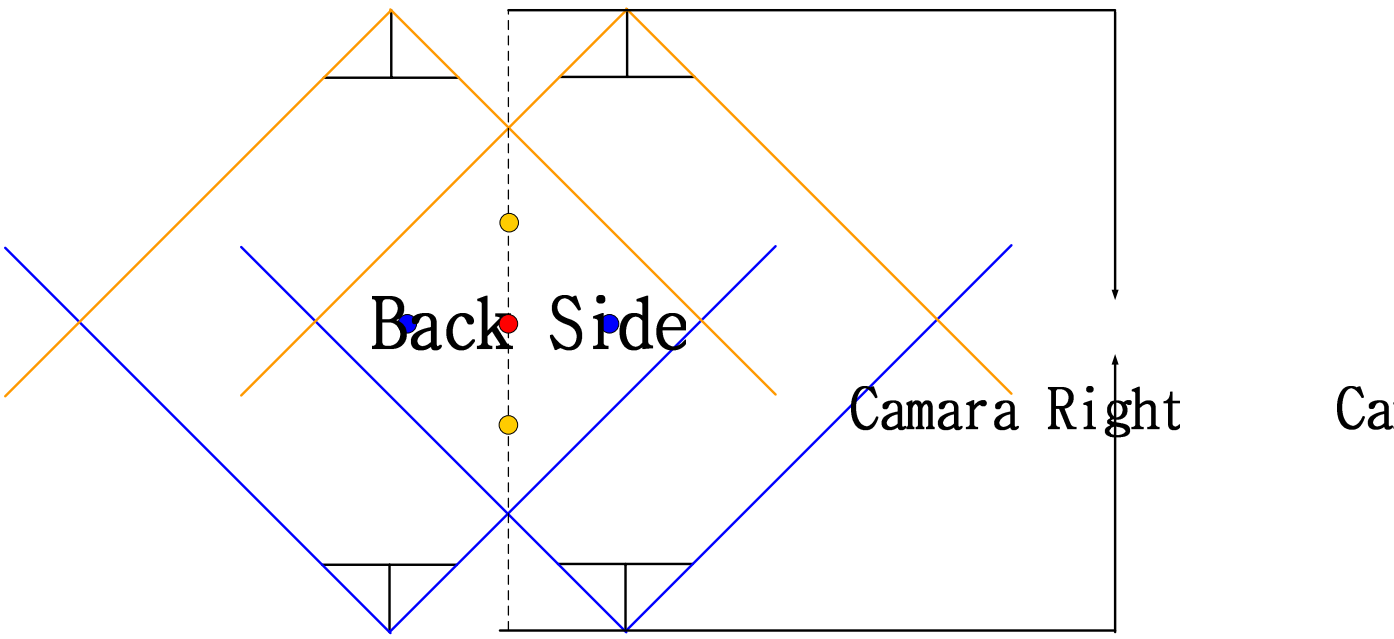
รูปที่ 3.3 แสดงระบบการทำงานในการจับภาพการเคลื่อนไหว

จากรูปแสดงการจับภาพการเคลื่อนไหว โดยที่ก่อนการจับการเคลื่อนไหวจะต้องมีการจัดสภาพแวดล้อมให้เหมาะสมต่อการจับภาพ มีการควบคุมแสงให้พอดี และพื้นที่ให้เพียงพอต่อการใช้งาน การรับสัญญาณภาพจากกล้องวิดีโอจะรับสัญญาณภาพผ่านเข้ามาทางการ์ดรับสัญญาณภาพ จากนั้นใช้โปรแกรม Capture Pro เพื่อทำการรับภาพจากกล้องทั้ง 4 เข้ามาแล้วส่งไปทำการหาตำแหน่งมาร์กเกอร์ไปประมวลผลต่อจึงต้องมีการติดตั้งระบบที่จะใช้ประมวลผลเตรียมไว้ด้วย

3.3.1 การติดตั้งกล้องถ่ายวิดีโอ

การติดตั้งกล้องนั้นควรจะเลือกใช้กล้องที่มีชนิดเดียวกัน รุ่นเดียวกัน เพื่อให้มีความสมดุลของค่าพารามิเตอร์ในกล้องให้ใกล้เคียงกันที่สุด และการจัดตำแหน่งของกล้องนั้น ควรจะวางให้ฉากรับภาพภายในกล้องนั้นขนานกัน และอยู่ห่างกันไม่มากเกินไป เพื่อสะดวกในการนำภาพที่ได้ไปคำนวณหาตำแหน่ง 3D ในขั้นตอนของ Stereopsis นั้นเอง ซึ่งในโครงงานนี้ใช้รูปแบบการวางกล้องแบบหันหน้า

ชนกันด้านละสองตัววางขนานกันซึ่งทั้งสองด้านต้องวางให้ห่างกันพอที่จะสามารถรับภาพผู้แสดงได้ทั่วทั้งตัวแล้วสามารถแสดงท่าทางต่างๆ ได้แต่ไม่ไกลจนเกินไป



รูปที่ 3.4 แสดงการวางตำแหน่งของกล้อง

เนื่องจากการวางกล้องแบบ ให้ Image plane นั้นอยู่ในระนาบเดียวกัน ซึ่งจะทำให้ลดขั้นตอนการคำนวณของ Stereopsis ได้มากและง่ายขึ้น การที่มีกล้อง 4 ตัวทำให้จะต้อง มีการหาค่าระยะห่างของกล้องด้านหน้ากับกล้องด้านหลัง(Distance_Z) เพื่อที่จะใช้ในการแปลงตำแหน่งจากกล้องด้านหลังให้เป็นตำแหน่งเดียวกันกับตำแหน่งกล้องทางด้านหน้า การหาค่าระยะห่างของกล้องด้านหน้ากับกล้องด้านหลัง(Distance_Z) นั้นจะต้องทำการกำหนดจุดขึ้นมาก่อน 1 จุดที่ตำแหน่งโดยจะเป็นตำแหน่งที่ตรวจจับมาได้เป็นตำแหน่งแรกของกล้องคู่หน้าและหลัง จากนั้นจะให้โปรแกรมทำการคำนวณค่าใจตำแหน่ง 3 มิติ และนำค่าแนวแกน Z ของกล้องด้านหน้าและกล้องด้านหลังบวกกันเพื่อหาระยะห่าง

ระหว่างกล้องคู่หน้าและกล้องคู่หลัง และนำไปกำหนดให้กับค่า Distance_Z Distance ซึ่งจำเป็นจะต้องนำไปใช้ในการแปลงตำแหน่งของ Z ของกล้องด้านหลัง มาเป็นด้านหน้าโดยใช้สมการดังนี้

$$Z_{front} = Distance_Z - Z_{back}$$

และค่าของ X จะต้องถูกแปลงด้วยเช่นกัน โดยการหาความแตกต่างของระยะในแนวแกน X โดยอาศัยจุดตำแหน่งเดียวกันกับการหา Distance_Z แต่จะพิจารณาค่าในแนวแกน X โดยใช้ตำแหน่งกล้องคู่หน้าเป็นตำแหน่งหลัก ลบด้วยตำแหน่งของกล้องคู่หลัง จะได้ค่า Distance_X และนำค่านี้ไปใช้ในการแปลงตำแหน่งแกน X ของกล้องคู่หลังมาเป็นตำแหน่งของกล้องคู่หน้า

$$X_{front} = Distance_X + X_{back}$$

ส่วนตำแหน่งของค่า Y นั้นคือความสูงทั้งสองกล้องจะมองตำแหน่งในลักษณะเดียวกันคือการหา Distance_Y จะเหมือนกันการหาในแนวแกน X โดยจะได้สมการในการแปลงตำแหน่งแนวแกน Y ของกล้องด้านหลังมาเป็นตำแหน่งของกล้องด้านหน้าดังนี้

$$Y_{front} = Distance_Y + Y_{back}$$

สาเหตุที่ทำให้ต้องใช้ marker ออกไปให้กล้องตรวจจับและหาระยะทางนั้นเนื่องจาก ระยะที่วัดจากจุด Focus ของกล้องทั้งสองนั้นเราไม่สามารถมองเห็นได้จึงจำเป็นต้องใช้วิธีดังกล่าว

3.4 ผู้แสดง มาร์กเกอร์ และการจัดสภาพแวดล้อม

ผู้แสดงนั้นเป็นส่วนหนึ่งในระบบ การคัดเลือกผู้แสดงนั้นไม่มีข้อกำหนดแต่อาจจะต้องมีการเลือกผู้แสดงที่มีรูปร่างเหมาะสม จะสามารถทำให้สามารถเคลื่อนไหวและแสดงท่าทางที่ต้องการได้ แต่ก็ขึ้นอยู่กับรูปแบบของตัวละครที่อยากจะให้แสดงออกมา โดยในการจับการเคลื่อนไหวของผู้แสดงนั้นระบบจะหาตำแหน่งข้อต่อสำคัญเพื่อให้สามารถนำไปสร้างการเคลื่อนไหวให้กับตัวละคร 3 มิติได้ ในการจะหาตำแหน่งข้อต่อของผู้แสดงจากภาพวิดีโอได้นั้นจำเป็นที่จะต้องใช้สัญลักษณ์ คือมาร์กเกอร์ (Marker) เป็นจุดสังเกต เพื่อให้โปรแกรมที่ใช้ประมวลผลภาพสามารถแยกแยะหาตำแหน่งของมาร์กเกอร์จากภาพ ดังนั้นโครงงานนี้ใช้มาร์กเกอร์ที่มีคุณสมบัติสะท้อนแสง หรือสามารถให้แสงสว่างด้วยตัวเองได้นั้นคือหลอดไฟเนื่องจากการประมวลผลภาพจะหาตำแหน่งของมาร์กเกอร์จากจุดที่มีความสว่างมากบนภาพ โดยมาร์กเกอร์จะติดไว้ตรงบริเวณที่จุดข้อต่อที่สำคัญต่างๆ ของร่างกาย สำหรับชุดที่ผู้แสดง (Actor) ต้องสวมใส่ จะเป็นชุดรัดรูปสีดำล้วน แขนยาว ขายาว เพื่อให้สามารถเคลื่อนไหวได้อย่างสะดวก และเพื่อไม่ให้เกิดการสะท้อนของแสง ซึ่งจะทำให้การคำนวณหาตำแหน่งคลาดเคลื่อนน้อยลง ลักษณะของสิ่งแวดล้อม ลักษณะของสิ่งแวดล้อมก็มีผลกับระบบ หากสถานที่ ที่ทำการจับภาพเป็นพื้นที่ ที่ไม่สามารถควบคุมแสงได้ก็จะก่อให้เกิดความลำบากต่อการสร้างระบบและส่งผลกระทบ

ต่อการประมวลผลภาพที่รับเข้าไป ดังนั้นสถานที่ ควรจะเป็นพื้นที่เปิด และไม่มีแสงสว่างมากเกินไป เช่นแสงจากแสงอาทิตย์

3.4.1 มาร์กเกอร์(Marker)

มาร์กเกอร์ที่ใช้เป็นหลอดไฟฉาย หรือหลอดไฟท้ายรถจักรยานยนต์หลอดเล็กๆ เลือกที่กินไฟน้อยหาซื้อได้ทั่วไปตามท้องตลาด สามารถนำมาติดใกล้ตัวและไม่เป็นอันตรายถึงชีวิตในโครงงานนี้ใช้หลอดไฟ 6V นำหลอดมาต่อกันให้สามารถนำไปติดได้ทั่วทุกจุดที่ต้องการ

รูปที่ 3.5 หลอดไฟที่นำมาใช้สำหลับทำเป็นมาร์กเกอร์

จุดมาร์กเกอร์ 21 จุดหลัก

1. เป็นจุดบริเวณข้างบนศีรษะกึ่งกลาง
2. เป็นจุดบริเวณหัวไหล่ด้านขวา
3. เป็นจุดบริเวณหัวไหล่ด้านซ้าย
4. เป็นจุดบริเวณกึ่งกลางหน้าอกด้านหน้า
5. เป็นจุดบริเวณข้อศอกด้านขวา
6. เป็นจุดบริเวณข้อศอกด้านซ้าย
7. เป็นจุดบริเวณข้อมือขวาด้านนอกลำตัว
8. เป็นจุดบริเวณข้อมือซ้ายด้านนอกลำตัว
9. เป็นจุดบริเวณกลางหลังมือขวา
10. เป็นจุดบริเวณกลางหลังมือซ้าย

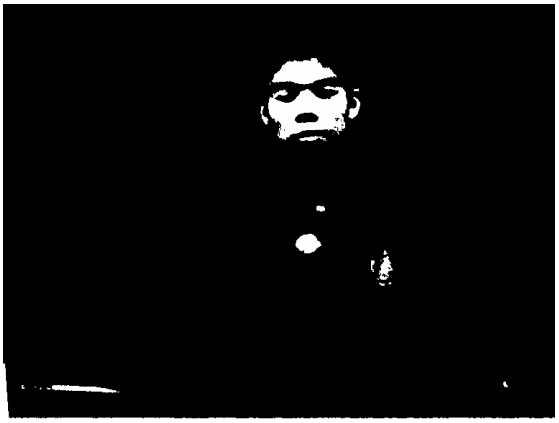
11. เป็นจุดบริเวณข้างสะโพกข้างด้านขวา
12. เป็นจุดบริเวณข้างสะโพกข้างด้านซ้าย
13. เป็นจุดบริเวณกลางสะโพกข้างด้านหน้า
14. เป็นจุดบริเวณด้านข้างหัวเข่าข้างขวา
15. เป็นจุดบริเวณด้านข้างหัวเข่าข้างซ้าย
16. เป็นจุดบริเวณตาตุ่มข้อเท้าขวาด้านนอก
17. เป็นจุดบริเวณด้านหน้าของเท้าขวา
18. เป็นจุดบริเวณตาตุ่มข้อเท้าซ้ายด้านนอก
19. เป็นจุดบริเวณด้านหน้าของเท้าซ้าย

สำหรับตำแหน่งที่ตรวจจับทั้ง 19 จุดนั้นเป็นจุดหลักที่เพียงพอต่อการตรวจจับการเคลื่อนไหวของร่างกายของมนุษย์ ซึ่งในการตรวจจับการเคลื่อนไหวแบบนี้สามารถเพิ่มและลดจำนวนมาร์กเกอร์ได้ไม่ยากนักจึงอำนวยต่อการปรับเปลี่ยนมาร์กเกอร์

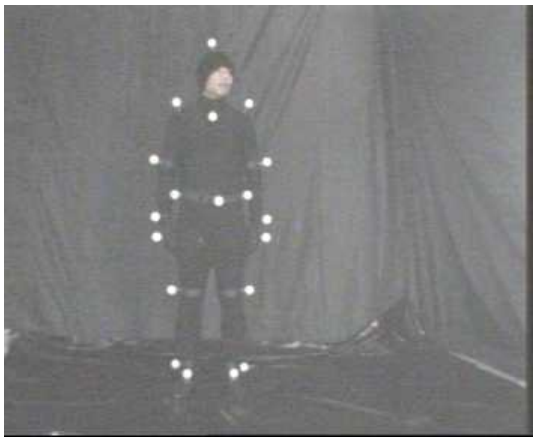
3.4.2 สภาพแวดล้อม

สภาพแวดล้อมก็มีส่วนสำคัญต่อการตรวจจับ เพราะหากจับภาพในที่ที่มีแสงมากเกินไปก็อาจทำให้ส่วนอื่นที่ไม่สนใจมีสภาพเป็นสีขาว หรือหากถ้ามืดเกินไปก็อาจจะออกเป็นสีดำดังตัว ดังนั้นจึงต้องจัดสภาพแสดงให้พอดี โดยจากการทดลองพบว่าการตรวจจับในห้องที่ไม่แสง มีความเหมาะสมที่สุด เนื่องจากจะสามารถควบคุมได้ง่าย

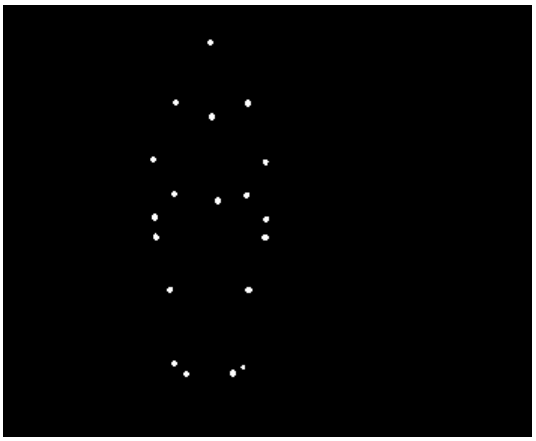
(ก)



(ข)



(ค)



(ง)

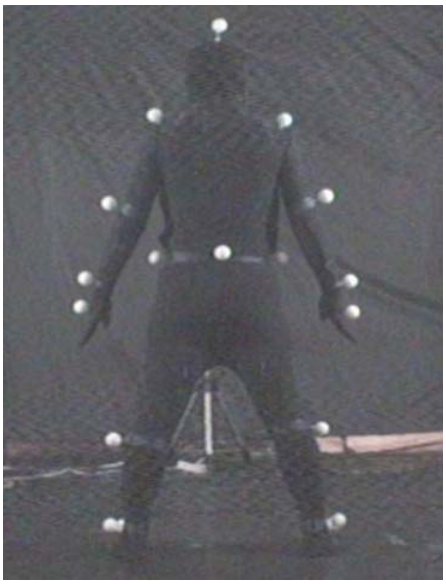
รูปที่3.6 (ก) รูปการจับภาพในสภาวะแวดล้อมที่มีแสงมากเกินไป

*(ข) รูปผลจากการทำ **Thresholding** ของสภาวะแวดล้อมที่มีแสงมากเกินไป*

(ค) รูปการจับภาพในสภาวะแวดล้อมที่มีแสพอดี

*(ง) รูปผลจากการทำ **Thresholding** ของสภาวะแวดล้อมที่มีแสงพอดี*

(ก)

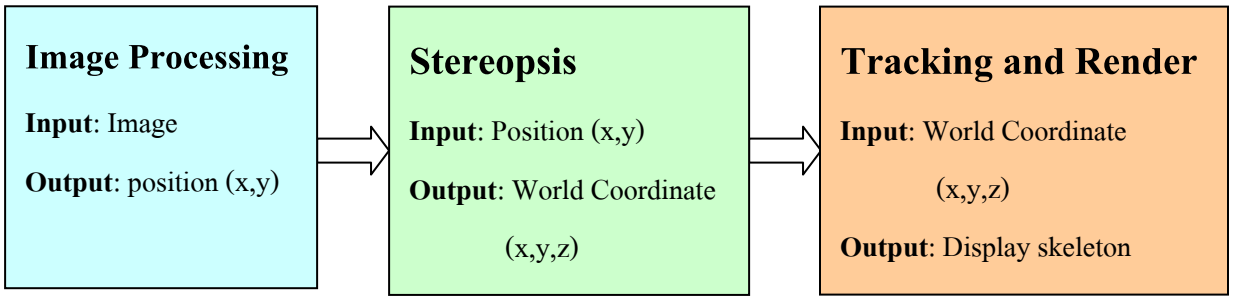


(ข)

รูปที่ 3.7 (ก) ตำแหน่งมาร์กเกอร์ด้านหน้าของผู้แสดง

(ข) ตำแหน่งมาร์กเกอร์ด้านหลังของผู้แสดง

3.5 การวิเคราะห์หาตำแหน่งและการเคลื่อนไหว (Motion Analysis)

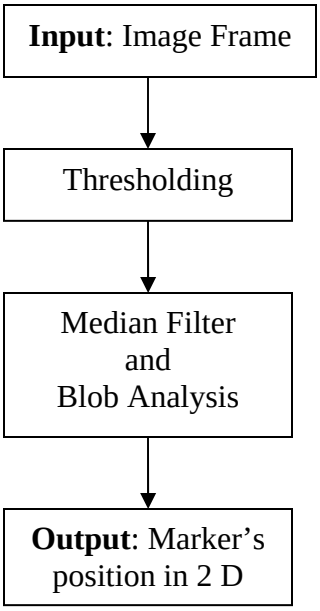


รูปที่ 3.8 การวิเคราะห์หาตำแหน่งและการเคลื่อนไหว (Motion Analysis)

การวิเคราะห์หาตำแหน่งและการเคลื่อนไหวจะเริ่มหลักจากได้รับภาพมาจากส่วนรับภาพมา ก็ส่งต่อมาในส่วนของการประมวลผลภาพผลลัพธ์ที่ได้คือตำแหน่งของมาร์กเกอร์แต่ละอันบนภาพแต่ละภาพ เสร็จจากส่วนนี้ก็จะนำไปเป็นอินพุทให้กับส่วนของการหาตำแหน่งจริง (World Coordinate)โดยใช้การคำนวณจากทฤษฎี Stereopsis จากนั้นก็นำไปทำ Tacking and Render ซึ่งเป็นส่วนที่จะทำให้รู้ว่ามาร์กเกอร์แต่ละตัวมีการเคลื่อนที่จากจุดแรกไปจุดทีสองและจุดต่อๆไปในลักษณะใดเพื่อไปกำหนดให้ส่วนแสดงผล (Render) ทำให้โครงสร้างโครงกระดูกแสดงการเคลื่อนไหวตามนั้น

3.5.1 การประมวลผลภาพ(Image Processing)

หน้าที่หลักๆ ของการทำงานส่วนในนี้คือการหาตำแหน่งของมาร์กเกอร์แต่ละตัวที่อยู่บนภาพที่รับเข้ามา



รูปที่ 3.9 แผนผังแสดงการประมวลผลภาพเพื่อหาตำแหน่งใน 2 มิติของมาร์กเกอร์

3.5.1.1 Thresholding

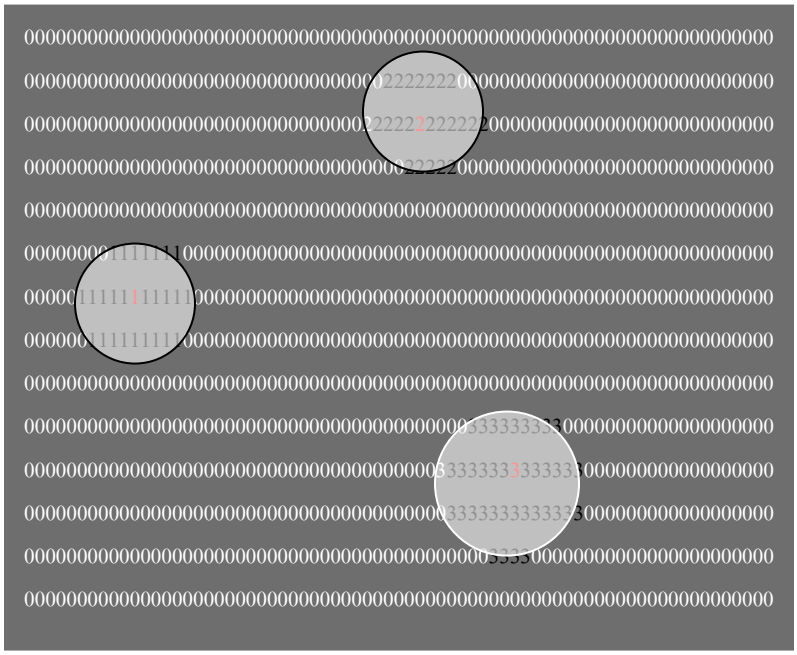
การทำ Threshold ในโครงงานนี้จะทำ Threshold โดยจะเลือกเอาส่วนใดๆของภาพที่มีลักษณะเป็นสีขาวและมีค่าความสว่างมาก (คือส่วนที่เป็นมาร์กเกอร์) ในที่นี้จะใช้ค่า V (value,intensity) ในระบบสี HSV ซึ่งเป็นค่าความสว่างในการตรวจสอบ ที่ตำแหน่งไหนมีค่า V มากตามต้องการคือประมาณไม่น้อยกว่า 90% ก็แสดงว่าตำแหน่งนั้นคือส่วนที่เป็นมาร์กเกอร์ ก็ให้ทำให้ ณ พิกเซลมีค่าเป็นบิต 1 ที่เหลือที่ไม่ตรงตามเงื่อนไขก็ให้เป็นบิต 0 ตัวอย่าง

3.5.1.2 Median Filter and Blob Analysis

หลังจากทำThresholdเราจะได้จุดสีขาวที่เป็นมาร์กเกอร์หลายๆอัน ซึ่งอาจจะมีจุดสีขาวที่เราไม่สนใจเนื่องมาจากเป็นจุดที่ไม่ได้เกิดมาจากมาร์กเกอร์ จึงต้องกำจัดก่อนโดยการทำ Median Filter ภาพจากการทำ Threshold และ Median Filter เสร็จแล้วจะเป็นภาพที่แยกตัวมาร์กเกอร์ที่เจอกับพื้นหลังอย่างชัดเจน คือส่วนที่สนใจจะเป็นบิต 1 แล้วส่วนที่ไม่สนใจจะเป็นบิต 0 จากนั้นก็นำมาทำการระบุหาตำแหน่งของมาร์กเกอร์แต่ละตัวที่ตรวจเจอให้เหลือเพียงพิกัดเดียว โดยวิธีการหาตำแหน่งมาร์กเกอร์แต่ละตัวนี้จะใช้วิธีหาค่ากลางของ พิกัด x ทุกพิกเซลและพิกัด y ทุกพิกเซลของจุดสีขาวทุกตัวที่เจอในขอบเขตของมาร์กเกอร์เลือกมาเป็นพิกัดบนภาพของมาร์กเกอร์ตัวนั้นๆ

ขั้นตอนการทำงาน Blob Analysis

1. ระบุหมายเลขให้ทุกพิกเซลที่อยู่ในขอบเขตของมาร์กเกอร์ตัวที่หนึ่งให้เป็น 1 ให้เหมือนกันทุกตัว
2. ทำตามข้อแรกกับมาร์กเกอร์ตัวต่อไปโดยให้เปลี่ยนหมายเลขเป็น 2 และทำไปกับมาร์กเกอร์ตัวต่อๆ โดยเพิ่มเลขที่ระบุไปเรื่อยๆจนครบทุกตัว ฉะนั้นเมื่อเสร็จขันตอนนี้ก็จะได้ดังรูปที่ 3.10



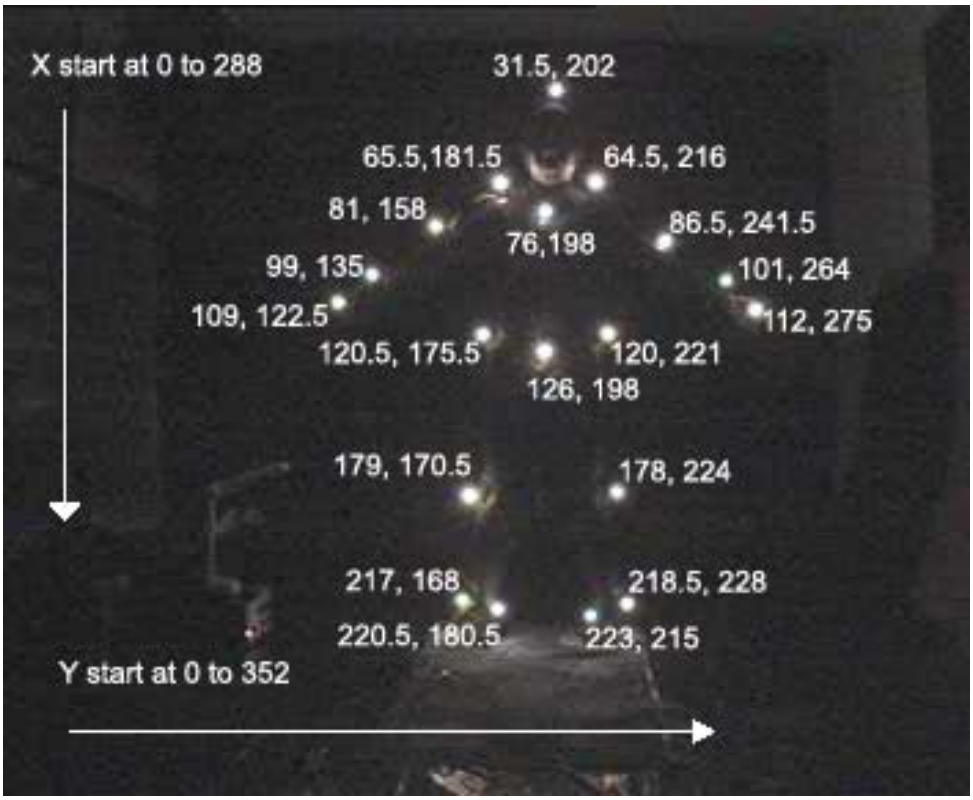
รูปที่ 3.10 การทำ Blob Analysis

3. หาค่ากลางของ พิกัด X ทุกพิกเซลและพิกัด Y ทุกพิกเซลในแต่ละกลุ่มหมายเลขที่ระบุไว้
4. กำหนดให้พิกัด X ทุกพิกเซลและพิกัด Y เป็นพิกัดตำแหน่งของมาร์กเกอร์นั้นๆ

เมื่อเสร็จขั้นตอนนี้ผลลัพธ์ที่ได้ก็คือ ตำแหน่ง X และ Y ที่เลือกแล้วของมาร์กเกอร์แต่ละตัวของภาพที่พิจารณาอยู่ภาพและก็จะส่งต่อไปให้ฟังก์ชัน Stereopsis หาตำแหน่งจริง (World Coordinate) ต่อไป

3.5.2 การระบุค่าเริ่มต้นให้กับมาร์กเกอร์(Initial post)

เนื่องจากโครงงานนี้ใช้มาร์กเกอร์เป็นชนิดเดียวกันหมดจึงทำให้ในแต่ละครั้งที่จับภาพออกมามาร์กเกอร์บนภาพจะมีลักษณะไม่แตกต่างกันจึงจำเป็นต้องระบุด้วยว่าตัวไหนเป็นตำแหน่งมาร์กเกอร์ที่เป็นส่วนหัว ส่วนไหล ส่วนอก หรือข้อเท้า โดยหลักการที่ใช้ระบุค่าเริ่มต้นคือ เก็บพิกัดของมาร์กเกอร์ทุกตัวมาแล้วเอามาหาเทียบว่าส่วนที่สูงที่สุดจะเป็นอวัยวะส่วนไหนโดยตรวจสอบจากค่าพิกัดทางแกน Y ที่มีค่ามากที่สุดเมื่อได้ตำแหน่งที่มีค่าพิกัด Y ที่มากที่สุดแล้วก็ให้เป็นตำแหน่งหัวเมื่อได้แล้วก็ตัดตำแหน่งนั้นออกแล้วเริ่มทำการหาใหม่โดยรอบต่อมาจะเป็นส่วนไหล่ก็ต้องหาตำแหน่งพิกัด Y มากที่สุดและรองลงมารวมเป็นสองตำแหน่งเพราะว่าไหลสองข้างอาจจะมีความสูงใกล้เคียงกันจึงต้องรับมาสองตำแหน่งเพื่อมาหาว่าตำแหน่งไหนเป็นไหล่ซ้าย และไหล่ขวาโดยรู้ได้จากค่าพิกัดทางด้านแกน X เปรียบเทียบกันด้านที่มีค่ามากกว่าก็จะเป็นอวัยวะทางด้านซ้ายและด้านที่มีพิกัดทางด้านแกน X น้อยกว่าจะเป็นด้านขาว ส่วนอวัยวะอื่นๆก็ใช้แนวคิดเดียวกันนี้ ฉะนั้นจึงต้องมีการกำหนดท่าทางในการทำ Initial post ให้เป็นมาตรฐานเพื่อให้โปรแกรมสามารถคำนวณผลลัพธ์ออกมาได้อย่างถูกต้อง



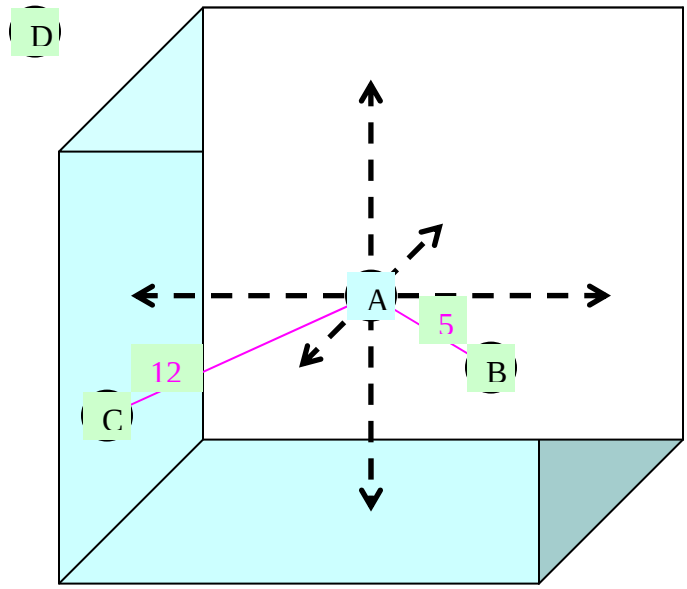
รูปที่ 3.11 การระบุค่าเริ่มต้นให้กับมาร์กเกอร์ (Initial post)

3.5.3 การติดตามมาร์กเกอร์

การติดตามตำแหน่งของมาร์กเกอร์แต่ละตัวจะทำหลังจากการหาค่าตำแหน่งของมาร์กเกอร์ในมิติที่สามได้แล้วคือหลังจากได้ค่าพิกัด (x , y) แล้วใช้ส่วน Stereopsis หาตำแหน่งใน 3 มิติ ก็จะได้ค่าพิกัดทั้งสาม (x, y, z) จากนั้นเมื่อมาร์กเกอร์มีการเปลี่ยนตำแหน่งไปจากเดิมก็จะทำการเอาตำแหน่งใหม่ทุกตำแหน่งมาเปรียบเทียบกับเดิมที่กำลังพิจารณาว่ามีตำแหน่งไหนอยู่ในรัศมีใกล้กลับตำแหน่งที่กำลังพิจารณาอยู่ก็ให้เป็นตำแหน่งใหม่ของตำแหน่งที่กำลังพิจารณาอยู่เลยโดยรัศมีจะใช้ขึ้นอยู่กับความความเร็วที่สามารถจับภาพได้

ขั้นตอนการทำงาน

1. ให้ค่าตั้งต้นเท่ากับรับค่ามาจากการทำ Initial post
2. รับค่าที่เกิดจากการเคลื่อนไหวมา
3. เปรียบเทียบค่าที่ได้จากการเคลื่อนไหวกับค่าตั้งต้น
4. แทนค่าที่เกิดจากการเคลื่อนไหวให้กับค่าตั้งต้นตัวที่พิจารณาอยู่นั้นถ้าหากค่าที่เกิดจากการเคลื่อนไหวอยู่ในรัศมีของค่าตั้งต้น

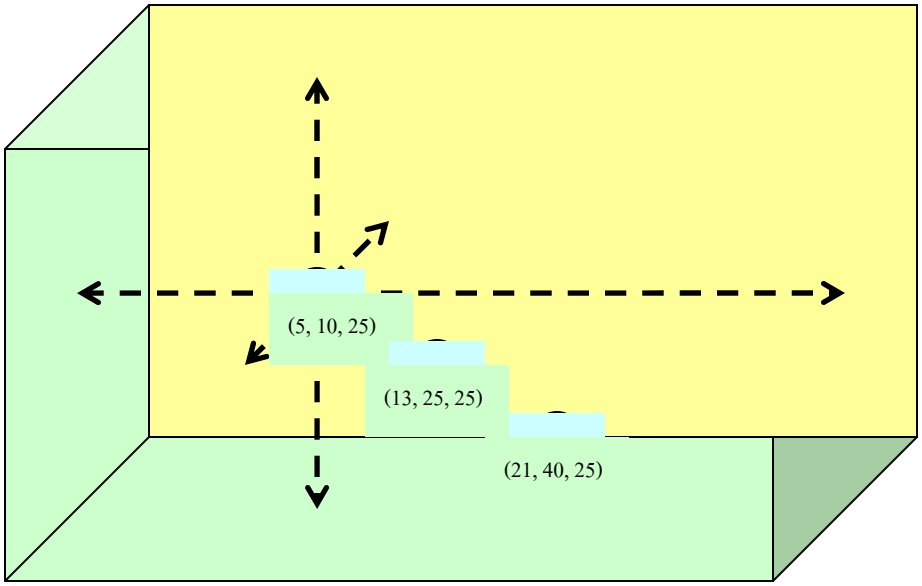


รูปที่ 3.12 การติดตามมาร์กเกอร์

จากรูป 3.12 จุด A เป็นตำแหน่งก่อนหน้าการ Track จะยึดเอาจุด A เป็นจุดอ้างอิงในการเปรียบเทียบหาตำแหน่งถัดไปของตัวมันเองถ้าหากพบจุดที่ค่า X ค่า Y และค่า Z ทั้งสามอยู่ในรัศมีก็จะถือว่าจุดนั้นเป็นจุดถัดไปของตัวมันเอง ในกรณีที่พบจุดที่อยู่ในรัศมีมากกว่าสองจุดก็จะทำการหาระยะห่างจากจุดอ้างอิงกับจุดทั้งหมดที่เจอแล้วเลือกเอาจุดที่มีระยะห่างน้อยที่สุดเป็นตำแหน่งถัดไปของจุดอ้างอิงจากรูป ก็จะเลือกเอาจุด B เป็นตำแหน่งถัดไปของ A เพราะว่าระยะห่างจาก A ไป B น้อยกว่าระยะห่างจาก A ไป C

3.5.4 การแก้ไขเมื่อตรวจไม่พบตำแหน่งถัดไป

การแก้ไขเมื่อตรวจไม่เจอตำแหน่งถัดไปจากตำแหน่งของมาร์กเกอร์ทั้งหมดที่หามา ในกรณีนี้โปรแกรมได้มีการเก็บตำแหน่งก่อนหน้าของตำแหน่งก่อนไว้ โปรแกรมจะทำการหาว่าก่อนหน้านี้ตำแหน่งที่สนใจนี้มีการเคลื่อนที่จากตำแหน่งก่อนหน้าไปทิศทางละเท่าใด เช่นค่า X มีการเคลื่อนที่ไป 8 พิกเซลค่าY มีเคลื่อนที่ไป 15 ส่วนค่า Z ไม่มีการเคลื่อนที่ก็จะเป็น 0 ก็จะนำค่าเหล่านี้ไปบวกเพื่อหาตำแหน่งใหม่ ดังรูป 3.13



รูปที่ 3.13 การแก้ไขเมื่อตรวจไม่พบตำแหน่งถัดไป

จากรูป3.13 เมื่อ A2 หาตำแหน่งถัดไปไม่เจอจากตำแหน่งทั้งหมดที่ให้มา โปรแกรมก็จะเอาระยะทางที่เคลื่อนที่จาก A1 ไปยัง A2 มาบวกเพิ่มให้กับ A2 ก็จะได้ตำแหน่ง A3 เป็นตำแหน่งใหม่ โดยทั่วไปแล้วมีวิธีที่ดีกว่านี้ในการหาตำแหน่งถัดไปที่หาไม่เจอ เช่น การคำนวณหาโดยทฤษฎี Inverse kinematic จะใช้การคำนวณโดยอ้างอิงตำแหน่งอวัยวะที่พิจารณาอยู่กับอวัยวะใกล้เคียงแต่เนื่องด้วยวิธีเหล่านั้นล้วนแล้วใช้เวลาในการคำนวณเป็นอย่างมากจึงไม่เหมาะสมกับการทำงานแบบ Real time ดังนั้นโครงงานชิ้นนี้นำการทำงานง่ายๆ มาใช้งานแทน

3.6 ระบบประมวลผลที่ใช้

ระบบสร้างตัวละคร 3 มิติโดยการตรวจจับการเคลื่อนไหวของร่างกายเป็นระบบที่ทำงานแบบ Real time ซึ่งการคำนวณที่ล่าช้าและเกิด Delay time มากๆจะมีผลกระทบต่อระบบโดยรวมได้ เครื่องคอมพิวเตอร์ที่นำมาใช้ในการประมวลผลก็มีส่วนกับความเร็วของการประมวลผลเช่นกัน ฉะนั้นในโครงงานชิ้นนี้จึงได้ใช้เครื่องคอมพิวเตอร์สองเครื่องแยกกันประมวลผลแล้วส่งผลลัพธ์สุดท้ายที่เป็นตำแหน่งจริง (World Coordinate) มาให้เครื่องใดเครื่องหนึ่งเป็นตัวแสดงการเคลื่อนไหวตัวละคร 3 มิติ

System:

Microsoft Windows XP Professional, Version 2002

Service Pack 2

Computer:

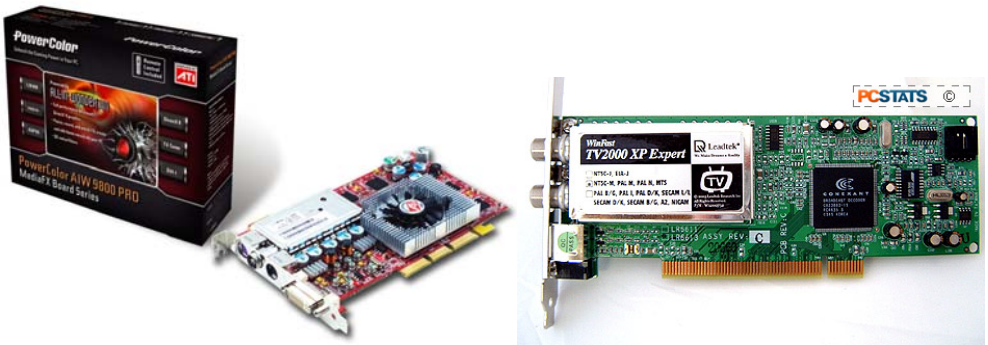
Intel® Pentium® 4 CPU 2.80 GHz 2.7 GHz

RAM 512 MB

Video Capture Cards:

All-In-Wonder Series PowerColor AIW 9800 PRO

Winfast TV2000 XP Expert Card



รูปที่ 3.15 Capture cards