สเตอริโอวิชั่น

Stereo Vision

โดย

นายชวลิต ทวีโชคสมบัติ

นายธนัญชัย โกสินตระกูลชัย

นางสาวลลิตา อินทราลักษณ์

วัน เดือน ปี 14 พ.ค. 2539

เลขทะเบียน 034729

เลขเรียกหนังสือ T 57.5 ว5

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

034729

ปริญญานิพนธ์ ปีการศึกษา 2537

ภาควิชา วิศวกรรมระบบควบคุม

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหาร ลาดกระบัง

เรื่อง สเตอริโอวิชั่น

Stereo Vision

ผู้จัดทำ นายชวลิต ทวีโชคสมบัติ 34102081

นายธนัญชัย โกสินตระกูลชัย 34103136

นางสาวลลิตา อินทราลักษณ์ 34105293

( )

อ.พรสุข เทศเจริญ

( )

อ.เกษตร์ ศิริสันติสัมฤทธิ์

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ สเตอริโอวิชั่น

Stereo Vision

โดย นายชวลิต ทวีโชคสมบัติ

นายธนัญชัย โกสินตระกูลชัย

นางสาวลลิตา อินทราลักษณ์

อาจารย์ที่ปรึกษา อ.พรสุข เทศเจริญ

อ.เกษตร์ ศิริสันติสัมฤทธิ์

บทคัดย่อ

การหาระยะทางโดยใช้ภาพ (*image*) ของวัตถุซึ่งถูกตรวจจับโดยกล้อง CCD นั้นเป็นการศึกษาปัจจัยพื้นฐานในการมองเห็นของหุ่นยนต์ (*Robot Vision*) ในโครงงานนี้จะใช้วิธี *Zero-crossing* หาขอบเขตของภาพของวัตถุ จากนั้นจึงหาความสัมพันธ์ระหว่างภาพทั้งสองภาพเพื่อหาจุดที่เหมือนกัน (*matching point*) และสุดท้ายจะสามารถหาระยะทางได้จากสมการความสัมพันธ์ของเลนส์และระบบพิกัด (*Image Geometry*) โดยเปรียบเทียบกับระยะทางจริงที่วัดได้จากการทดลองแล้วจึงนำมาปรับปรุงสมการหาระยะทางต่อไป จนกว่าได้ระยะทางที่ใกล้เคียงกับระยะทางจริง

Abstract

The object distance determination using captured images of observation object by CDD camera models is the basic study of seeing ability for Robot vision . In this project , the Zero-crossing method is chosen for edge detection , then find the relation of pixels in each image to obtain the match point . Thus compose with lens equations , and find image geometry equation . Interpolate the real measured distance and the improved equation to decrease the probability error until the observed distance is closed to the real one.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบคุณอาจารย์และเพื่อน ๆ ทุกคนที่มีส่วนช่วยให้ปริญญานิพนธ์นี้สำเร็จลุล่วงไปด้วยดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทนำ **สเตอริโอวิชั่น** (1)

บทที่ 2 **ระบบพิกัดจำลองกับการคำนวณหาระยะทาง** (4)

บทที่ 3 **Matching point method** (12)

- Zero-crossing of Laplacian - Guassian Filtered Image (12)
- Detection of Intensity Changes (13)
- อัลกอริทึมพื้นฐานในการกำหนดตำแหน่งขอบและทิศทางของภาพ (15)
- อัลกอริทึมในการหาจุดที่เหมือนกัน (16)

บทที่ 4 **การทดลองหาระยะทาง** (17)

- ขั้นตอนการทดลอง (17)
- ผลการทดลองหาค่า K (19)
- ผลการทดลองหาระยะทาง (21)
- สรุปผลการทดลอง (23)
- วิจารณ์ผลการทดลอง (23)

หนังสืออ้างอิง (24)

ภาคผนวก ก. (25)

ภาคผนวก ข. (81)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

- ตารางที่ 4.1 การหาค่า K (20)
- ตารางที่ 4,2 ผลการทดลอง กรณีที่ 1 (21)
- ตารางที่ 4.2 ผลการทดลอง กรณีที่ 2 (22)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

- รูปที่ 1.1 แสดงการวางตำแหน่งกล้องสองกล้องกับวัตถุหนึ่งชิ้น (2)
- รูปที่ 1.2 แสดงภาพที่เกิดจากกล้องทั้งสอง (2)
- รูปที่ 2.1 แสดงความสัมพันธ์ระหว่าง *world coordinate* และ *camera coordinate* (4)
- รูปที่ 2.2 แสดงระบบพิกัดของวัตถุและกล้องตามตัวอย่าง (5)
- รูปที่ 2.3 แสดงการแปลงพิกัดของระบบให้เป็นไปตามต้องการ (8)
- รูปที่ 2.4 แสดงระบบพิกัด , กล้องและวัตถุ (9)
- รูปที่ 2.5 แสดงระบบพิกัดของภาพที่ถ่ายได้จากกล้อง (11)
- รูปที่ 3.1 แสดงพารามิเตอร์ของ LoG Masks (13)
- รูปที่ 3.2 แสดงผลตอบสนองของ LoG Masks (14)
- รูปที่ 4.1 ขาตั้งและกล้องที่ใช้ในการทดลอง (17)
- รูปที่ 4.2 แสดงการเซ็ตระบบเพื่อหาค่า K (18)
- รูปที่ 4.3 แสดงภาพวัตถุที่เก็บผ่านการ์ดดิจิไตเซอร์ (20)
- รูปที่ 4.4 แสดงการทำ Zero-crossing กับภาพ (19)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

Stereo Vision

การศึกษาเกี่ยวกับ *Stereo Vision* มีเนื้อหาที่กว้างมากแล้วแต่ว่าจะเน้นไปในเรื่องใดโดยเฉพาะ สำหรับปริญญานิพนธ์ฉบับนี้จะเป็นการศึกษา *Stereo Vision* **โดยเน้นการหาระยะทางระหว่างกล้องกับวัตถุ** ซึ่งกล้องที่ใช้ในที่นี้จะเป็นกล้อง CCD สองตัว (รายละเอียดของกล้องอยู่ในภาคผนวก ก.)

จุดประสงค์ของการทำปริญญานิพนธ์ในครั้งนี้ ก็เพื่อต้องการหาระยะทางระหว่างวัตถุที่ต้องการกับตัวกล้อง โดยจะใช้กล้อง CCD สองตัวทำการวางอยู่ในระบบที่กำหนดขึ้นมา จากนั้นจะถ่ายภาพโดยใช้กล้องทั้งสอง ภาพที่ได้นี้เองจะนำไปคำนวณเพื่อที่จะหาระยะทางระหว่างกล้องกับวัตถุนั่นเอง

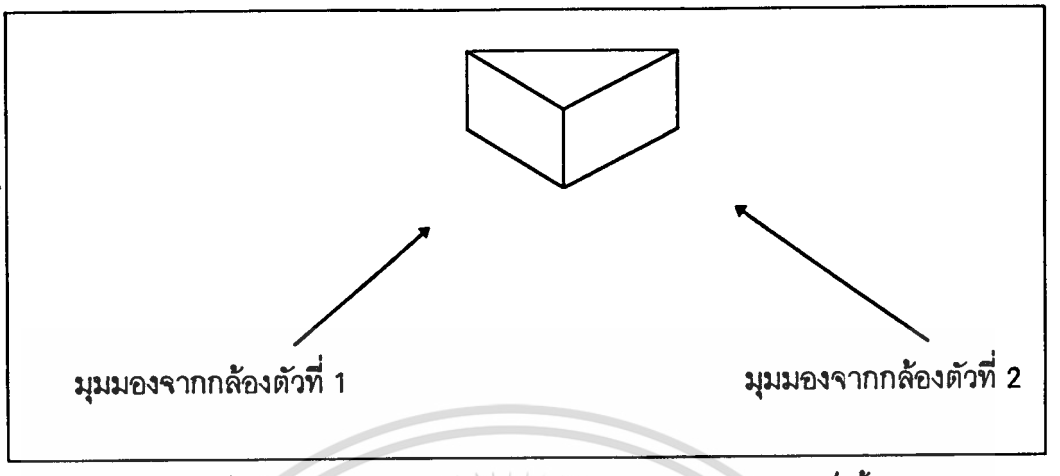
ดังนั้นเพื่อความง่ายและความเป็นระเบียบในการศึกษาเนื้อหาในบทต่อ ๆ ไปนั้น จึงควรแบ่งการทำงานออกเป็นขั้นตอนย่อย ๆ โดยจะกล่าวถึงปัญหาที่เกิดขึ้นในแต่ละขั้นตอนด้วย สำหรับรายละเอียดของแต่ละขั้นตอนจะกล่าวอย่างละเอียดอีกทีในบทต่อ ๆ ไป

สำหรับขั้นตอนการทำงานเราพอจะแบ่งได้ดังนี้

1.) กำหนดระบบพิกัดจำลองของระบบ (ระบบในที่นี้คือกล้องและวัตถุ) ขึ้นมาเสียก่อน โดยระบบพิกัดที่กำหนดขึ้นมานี้จะต้องสัมพันธ์กับสูตรหาระยะทางที่คิดขึ้นมา ดังนั้นปัญหาของขั้นตอนแรกนี้ก็จะเป็นปัญหาที่เกี่ยวกับการหาทฤษฎีคณิตศาสตร์เพื่อนำไปสู่การกำหนดพิกัดและสูตรหาระยะทางที่ถูกต้องแม่นยำต่อไป รายละเอียดของขั้นตอนนี้จะอยู่ในบทที่ 2

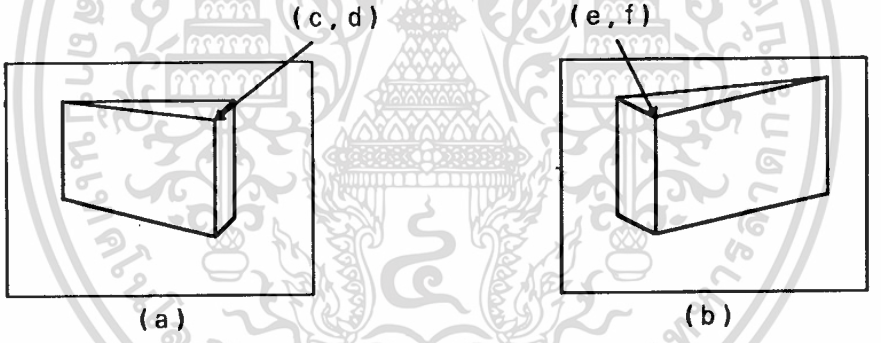
2.) นำภาพที่ได้จากกล้องทั้งสองมาหาจุดที่เหมือนกัน *(matching point)* ซึ่งความยากของขั้นตอนนี้ก็คือภาพทั้งสองที่ได้เป็นภาพของวัตถุเดียวกันที่ถ่ายจากกล้องคนละตัวในมุมมองที่ต่างกัน และการหาจุดที่เหมือนกันของภาพทั้งสองก็คือการหาตำแหน่งของวัตถุในรูปที่หนึ่งกับตำแหน่งของวัตถุเดียวกันในรูปที่สอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.1 แสดงการวางตำแหน่งกล้องสองกล้องกับวัตถุหนึ่งชิ้น

ตัวอย่างเช่นถ้าเราวางกล้องและวัตถุดังรูปที่ 1.1 ภาพที่ได้จากกล้องตัวที่ 1 และตัวที่ 2 จะเป็นดังนี้



รูปที่ 1.2 แสดงภาพที่เกิดจากกล้องทั้งสอง

รูปที่ 1.2 (a) คือภาพที่เกิดจากกล้องตัวที่ 1 และรูปที่ 1.2 (b) คือภาพที่เกิดจากกล้องตัวที่ 2 ถ้ากำหนดจุด (c , d) ในภาพที่ 1 เป็นจุดอ้างอิง เราจะได้จุดที่เหมือนกันกับจุด (c , d) ในภาพที่ 2 คือจุด (e , f)

ปัญหาสำคัญสำหรับขั้นตอนนี้ (และถือว่าเป็นปัญหาใหญ่สำหรับการทำปริญญานิพนธ์ครั้งนี้ด้วย) คือการหาจุดที่เหมือนกันในทางอุดมคตินั้นเอง เพราะในทางปฏิบัติแล้วแทบจะเป็นไปไม่ได้เลยที่จะหาจุดที่เหมือนกันได้อย่างถูกต้อง เนื่องภาพที่ถ่ายได้จะเต็มไปด้วยสัญญาณรบกวนอันจะทำให้การเปรียบเทียบหาจุดที่เหมือนกันทำได้ยากมาก อีกทั้งอัลกอริทึมที่คิดขึ้นมาเพื่อหาจุดที่เหมือนกันนี้ยังไม่มีอัลกอริทึมไหนให้ผลเป็นที่น่าพอใจ เช่น อาจต้องใช้เวลานานในการหา หรือ อาจจะได้จุดที่เหมือนกันโดยผิดพลาดจากความเป็นจริงมาก เป็นต้น (การหาจุดที่เหมือนกันนี้ในปัจจุบันยังเป็นหัวข้อที่ทำวิจัยกันอยู่)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับปริญญานิพนธ์นี้ได้เลือกวิธีการ *Zero Crossing* ในการหาจุดที่เหมือนกัน โดยรายละเอียดของวิธีนี้จะกล่าวอย่างละเอียดอีกทีในบทที่ 3

3.) นำพิกัดจุดที่เหมือนกันในภาพทั้งสองพร้อมทั้งตัวแปรต่าง ๆ ที่หาค่าได้แทนลงในสูตรหาระยะทาง ซึ่งผลที่ได้ก็จะคือระยะทางระหว่างกล้องกับวัตถุนั้นเอง แต่ปัญหาใหญ่สำหรับขั้นตอนนี้ดูจะเป็นปัญหาที่เกิดจากความผิดพลาดของมนุษย์เองเสียมากกว่า เช่น ความผิดพลาดที่เกิดจากการวัดหาค่าระยะต่าง ๆ ของตัวแปรผิดพลาด เป็นต้น

สำหรับประโยชน์ของปริญญานิพนธ์นี้อาจจะนำไปประยุกต์ใช้งานได้หลายอย่าง เช่น ตาหุ่นยนต์ หรือตารถสำรวจ เป็นต้น

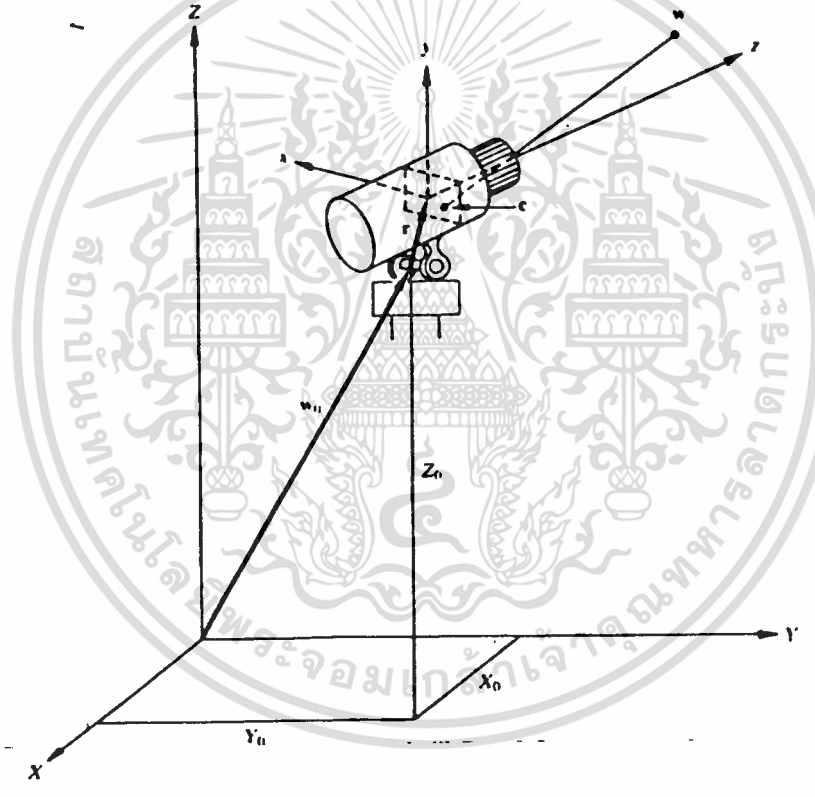
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ระบบพิกัดจำลองกับการคำนวณหาระยะทาง

เพื่อที่จะให้การวัดและการคำนวณต่าง ๆ เป็นไปอย่างถูกต้อง จำเป็นอย่างยิ่งที่จะต้องมีการกำหนดระบบพิกัดที่ใช้อ้างอิงขึ้นมาก่อน ซึ่งในที่นี้จะกำหนดพิกัดสำคัญ ๆ ขึ้นมา 2 พิกัด ก่อนนั้นคือ *world coordinate (X , Y , Z)* และ *camera coordinate (x , y , z)*

ดังนั้นเมื่อวัตถุอยู่ที่จุด *w* ใน *world coordinate* แล้ววัตถุดังกล่าวจะไปปรากฏอยู่ที่จุด *c* ใน *camera coordinate* ดังรูปที่ 2.1



รูปที่ 2.1 แสดงความสัมพันธ์ระหว่าง world coordinate และ camera coordinate

จากรูปที่ 2.1 จะเห็นว่ากล้องตั้งอยู่บนจุดหมุน ซึ่งสามารถหมุนเป็นมุม θ (เรียกว่ามุม *pan*) และมุม α (เรียกว่ามุม *tilt*) ซึ่งสามารถหมุนได้อย่างอิสระ เมื่อมุม θ คือมุมที่วัดระหว่างแกน *x* ใน *camera coordinate* กับแกน *X* ใน *world coordinate* และมุม α คือมุมที่วัดระหว่างแกน *z* ใน *camera coordinate* กับแกน *Z* ใน *world coordinate* โดยระยะทางที่สั้นที่สุดระหว่างจุดกำเนิดของ *world coordinate* ไปยังจุดหมุดของกล้องกำหนดให้เป็นเวกเตอร์ W_0 และระยะทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระหว่างจุดหมุนของกล้องไปยังศูนย์กลางของ *image plane* แทนด้วยเวกเตอร์ r (ประกอบด้วยเวกเตอร์ r_1 , r_2 , r_3) ดังรูปที่ 2.1

การแปลงจุดกำเนิดของ *world coordinate* ไปเป็นจุดหมุนของกล้องอาจทำได้โดยใช้ *matrix G* ดังนี้

$$G=\begin{bmatrix}1 & 0 & 0 & -X_0\\ 0 & 1 & 0 & -Y_0\\ 0 & 0 & 1 & -Z_0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad (2.1)$$

หรืออาจกล่าวได้ว่าจุด homogeneous world w_h ซึ่งอยู่ในพิกัด (X_0 , Y_0 , Z_0) เมื่อทำการแปลงเป็นจุดกำเนิดของระบบพิกัดใหม่โดยการแปลง Gw_h

มุม θ จะวัดระหว่างแกน x ใน *camera coordinate* และแกน X ใน *world coordinate* โดยหมุนรอบแกน z ใน *camera coordinate* เมื่อกำหนดให้ทิศทางเป็นบวกเมื่อหมุนทวนเข็มนาฬิกา แต่ในภาวะปกติมุม θ นี้จะมีค่าเป็นศูนย์ ในการคำนวณจะใช้ *transformation matrix* R_θ ดังนี้

$$R_\theta=\begin{bmatrix}\cos\theta & \sin\theta & 0 & 0\\ -\sin\theta & \cos\theta & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad (2.2)$$

มุม α วัดจากแกน z ใน *camera coordinate* ไปแกน Z ใน *world coordinate* โดยหมุนรอบแกน x ใน *camera coordinate* กำหนดให้ทิศทางมีค่าเป็นบวกเมื่อหมุนทวนเข็มนาฬิกา สำหรับ *transformation matrix* ที่ใช้คำนวณคือ

$$R_\alpha=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos\alpha & \sin\alpha & 0\\ 0 & -\sin\alpha & \cos\alpha & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad (2.3)$$

เพราะฉะนั้นจะได้ว่าถ้าทำการหมุนระบบไปเป็นมุม θ และ α แล้ว จะต้องใช้ matrix R ในการแปลงจากระบบหนึ่งไปสู่ระบบหนึ่งดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$R=\begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta\cos\alpha & \cos\theta\cos\alpha & \sin\alpha & 0 \\ \sin\theta\sin\alpha & -\cos\theta\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

ระยะทางจากจุดหมุนถึงจุดศูนย์กลางภาพสามารถแทนได้โดยเวกเตอร์ r หรือสามารถเขียนเป็น *matrix* ได้ดังนี้

$$C=\begin{bmatrix} 1 & 0 & 0 & -r_1 \\ 0 & 1 & 0 & -r_2 \\ 0 & 0 & 1 & -r_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

เมื่อต้องการแปลงระบบ *world coordinate* และ *camera coordinate* ซึ่งเป็นระบบ 3 มิติไปเป็นระบบ 2 มิติสามารถทำได้โดยใช้สมการ 2. 6

$$c_h = PCRGw_h \quad (2.6)$$

เมื่อ

$$P=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{\lambda} & 1 \end{bmatrix} \quad (2.7)$$

$$w_h=\begin{bmatrix} kX \\ kY \\ kZ \\ k \end{bmatrix} \quad (2.8)$$

เมื่อนำค่าในแถวที่ 4 ของ c_h ในสมการที่ 2.6 ไปหารค่าในแถวที่ 1 และแถวที่ 2 ของ c_h จะได้ค่าต่าง ๆ ดังนี้

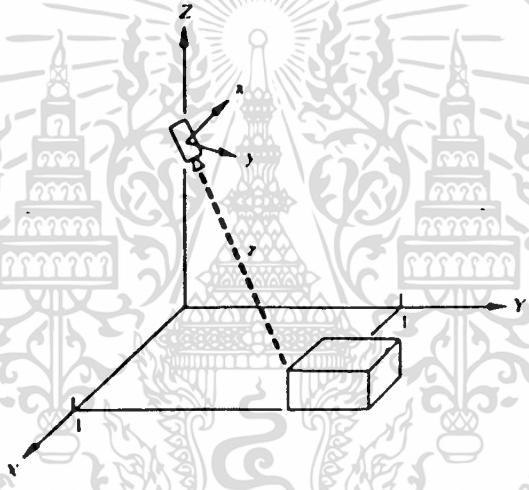
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$x = \lambda \frac{(X - X_0)\cos\theta + (Y - Y_0) - r_1}{-(X - X_0)\sin\theta\sin\alpha + (Y - Y_0)\cos\theta\sin\alpha - (Z - Z_0)\cos\alpha + r_3 + \lambda} \quad (2.9)$$

$$y = \lambda \frac{-(X - X_0)\sin\theta\cos\alpha + (Y - Y_0)\cos\theta\cos\alpha + (Z - Z_0)\sin\alpha - r_2}{-(X - X_0)\sin\theta\sin\alpha + (Y - Y_0)\cos\theta\sin\alpha - (Z - Z_0)\cos\alpha + r_3 + \lambda} \quad (2.10)$$

เพื่อให้เกิดความเข้าใจในระบบพิกัดที่กล่าวมาข้างต้นเป็นไปด้วยความชัดเจน จึงขอกล่าวถึงตัวอย่างการแปลงพิกัดดังตัวอย่างต่อไปนี้

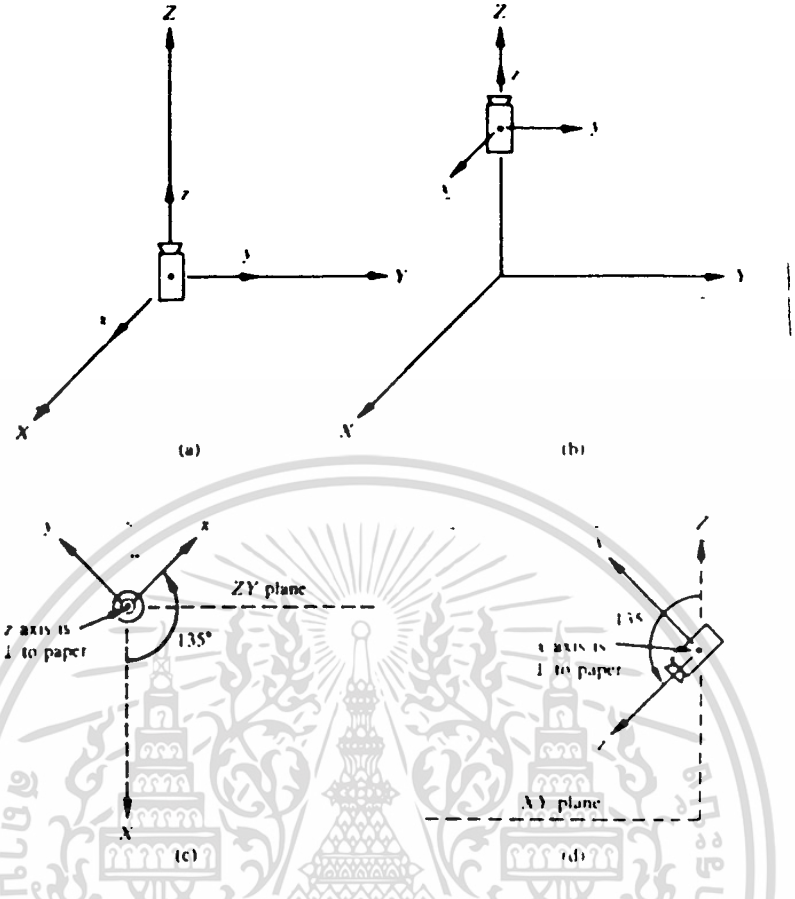
ตัวอย่าง ถ้าต้องการหา *image coordinaate* ของบริเวณมุมกล่องดังรูปที่ 2.2 โดยกล้องตั้งทำมุม θ เท่ากับ 135 องศา และมุม α เท่ากับ 135 องศา



รูปที่ 2.2 แสดงระบบพิกัดของวัตถุและกล้องตามตัวอย่าง

ในขั้นแรกจำเป็นที่จะต้องปรับกล้องจากตำแหน่งปกติไปอยู่ดังตำแหน่งในรูปที่ 2.3 เสียก่อน ซึ่งขั้นตอนในการปรับจะเป็นไปดังรูป (a), (b), (c) และ (d)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงการแปลงพิกัดของระบบให้เป็นไปตามต้องการ

รูปที่ 2.3 (a) แสดงตำแหน่งเมื่อกล้องวางอยู่ในตำแหน่งปกติ รูปที่ 2.3 (b) เป็นการเลื่อนกล้องขึ้นไปตามระยะทางที่ต้องการตามแกน Z รูปที่ 2.3 (c) แสดงการหมุนกล้องเป็นมุม θ (135 องศา) รอบแกน Z โดยจะหมุนตามเข็มนาฬิกา และรูปที่ 2.3 (d) แสดงการหมุนกล้องเป็นมุม α (135 องศา) ตามแกน X โดยจะหมุนตามเข็มนาฬิกา

จากรูปที่ 2.3 เมื่อทำการแปลงตำแหน่งของกล้องตามที่ต้องการแล้วจะได้ค่าต่าง ๆ ดังนี้

$$X_0 = 0$$
$$Y_0 = 0$$
$$Z_0 = 1$$
$$= 135^o$$
$$= 135^o$$
$$r_1 = 0.03$$
$$r_2 = r_3 = 0.02$$
$$\lambda = 0.035$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

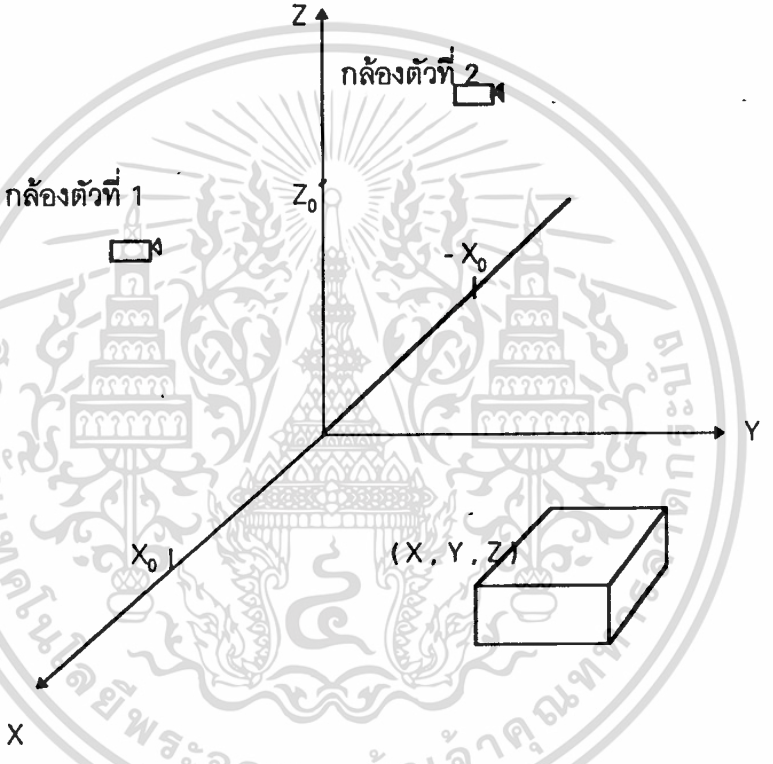
และมุมกล้องใน *coordinate* (X_0, Y_0, Z_0) = (1 , .1 , 0.2)

ดังนั้นจึงสามารถคำนวณมุมกล้องใน *image coordinate* ได้เท่ากับ

$$x = \lambda \frac{-0.03}{-1.53 + \lambda} = 0.0007$$

$$y = \lambda \frac{-0.42}{-1.53 + \lambda} = 0.009$$

ที่กล่าวมาข้างต้นเป็นการแนะนำระบบพิกัดที่กำหนดขึ้นมา ซึ่งในการใช้งานจริง ๆ จำเป็นที่จะต้องนำกล้องและวัตถุไปวางไว้ในระบบพิกัดดังกล่าวพร้อมทั้งหาความสัมพันธ์ทางคณิตศาสตร์เพื่อที่จะนำไปสู่การหาสูตรคำนวณหาระยะทางต่อไป



รูปที่ 2.4 แสดงระบบพิกัด , กล้องและวัตถุ

รูปที่ 2.4 แสดงระบบพิกัด , กล้องและวัตถุที่ใช้ในการทำปริญญานิพนธ์ครั้งนี้ และเพื่อให้เกิดความง่ายและถูกต้องในการทดลองจึงกำหนดให้กล้องวางในทิศทางขนานกับพื้นระนาบ ความยาวโฟกัสของกล้อง (λ) ทั้งสองที่ใช้เท่ากับ 2.8 มิลลิเมตร พร้อมกันนั้นกล้องทั้งสองจะต้องมีลักษณะทางกายภาพเหมือนทุกประการกันด้วย

ค่าตัวแปรต่าง ๆ ของกล้องตัวที่ 1 เป็นดังนี้ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
034729

$$X_0 = X_0$$
$$Y_0 = 0$$
$$Z_0 = Z_0$$
$$\theta = 180^0$$
$$\alpha = 90^0$$
$$r_1 = r_2 = r_3 = 0$$
$$\lambda = \lambda$$

และตัวแปรของกล้องตัวที่ 2 คือ

$$X_0 = -X_0$$
$$Y_0 = 0$$
$$Z_0 = Z_0$$
$$\theta = 180^0$$
$$\alpha = 90^0$$
$$r_1 = r_2 = r_3 = 0$$
$$\lambda = \lambda$$

จากสมการที่ 2.9 และ 2.10 เมื่อแทนค่าตัวแปรของกล้องตัวที่ 1 และตัวที่ 2 ลงไป โดยให้ x_1 และ y_1 แทนพิกัดของภาพที่ถ่ายได้จากกล้องตัวที่ 1 และ x_2 และ y_2 แทนพิกัดของภาพที่ถ่ายได้จากกล้องตัวที่ 2 เมื่อระบบพิกัดดังกล่าวเทียบกับภาพที่ถ่ายได้จะเป็นดังรูปที่ 2. 5

$$x_1 = \lambda \frac{-(X - X_0)}{-(Y - Y_0) + \lambda} \qquad (2.11)$$

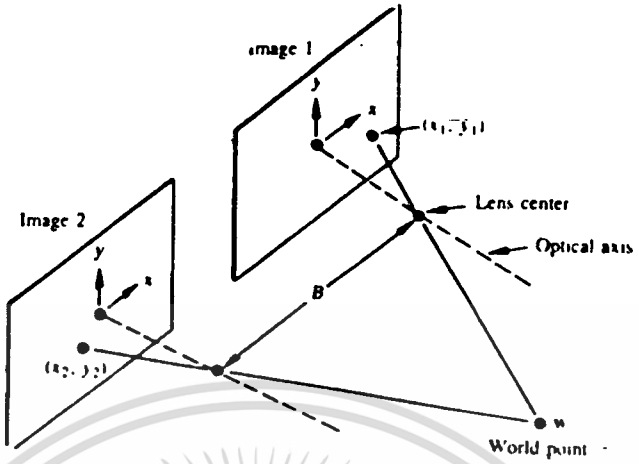
$$y_1 = \lambda \frac{(Z - Z_0)}{-(Y - Y_0) + \lambda} \qquad (2.12)$$

$$x_2 = \lambda \frac{-(X + X_0)}{-(Y - Y_0) + \lambda} \qquad (2.13)$$

$$y_2 = \lambda \frac{(Z - Z_0)}{-(Y - Y_0) + \lambda} \qquad (2.14)$$

จากสมการที่ 2.11 , 2.12 , 2.13 และ 2.14 ทำให้สามารถหาค่า (X , Y , Z) ซึ่งค่าเป็นพิกัดของวัตถุใน *world coordinate* ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2. 5 แสดงระบบพิกัดของภาพที่ถ่ายได้จากกล้อง

$$X = -\frac{X_0(x_1 + x_2)}{x_1 - x_2} \qquad (2.17)$$

$$Y = Y_0 + \lambda - \frac{2\lambda X_0}{x_1 - x_2} \qquad (2.16)$$

$$Z = \frac{2\lambda X_0 y_1}{x_1 - x_2} + Z_0 \qquad (2.17)$$

ดังนั้นระยะทางระหว่างกล้องตัวที่ 1 กับวัตถุจะเท่ากับ

$$d_1 = \sqrt{(X - X_0)^2 + (Y - Y_0)^2 + (Z - Z_0)^2} \qquad (2.18)$$

และระยะทางระหว่างกล้องตัวที่ 2 กับวัตถุเท่ากับ

$$d_1 = \sqrt{(X + X_0)^2 + (Y - Y_0)^2 + (Z - Z_0)^2} \qquad (2.19)$$

จากสมการความสัมพันธ์ทางคณิตศาสตร์ทั้งหมดที่กล่าวมาจะสรุปได้ว่าถ้าเราทราบค่าระยะห่างระหว่างกล้องทั้งสอง และค่าพิกัดของภาพที่ถ่ายได้จากกล้องทั้งสองแล้วเราจะสามารถหาระยะทางระหว่างกล้องและวัตถุได้ในที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

Matching point method

สำหรับเนื้อหาในบทนี้จะเน้นไปที่การหาจุดที่เหมือนกันของภาพสองภาพที่ถ่ายจะคนละมุมกล้อง โดยจะใช้วิธี Zero Crossing Edge Detection ซึ่งเป็นการประยุกต์วิธีหาขอบภาพ *(Egde Detection)* มาใช้นั้นเอง

การหาขอบภาพในปัจจุบันแบ่งเป็น 2 ประเภทใหญ่คือ

1.) วิธี *Gradient operators* เป็นผลตอบสนองของตำแหน่งสูงสุด ณ ตำแหน่งขอบภาพ อย่างไรก็ตามวิธีนี้สามารถใช้ได้ดีกับภาพที่มีการเปลี่ยนแปลงของขอบอย่างชัดเจนหรือมีขอบที่บางมาก

2.) วิธี *Second derivative operators* คือการหา *zero-crossing* ณ บริเวณพื้นที่ขอบซึ่งสามารถทำการ *interpolation* ระหว่างสัญญาณภาพกับสัญญาณรบกวนได้

Zero-crossing of Laplacian-Gaussian Filtered Images

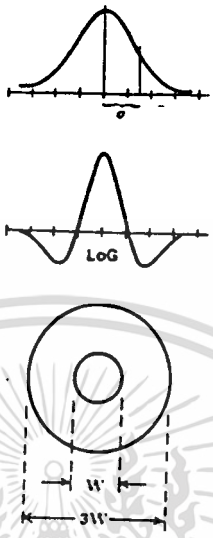
LoG convolution mask สามารถแสดงด้วยสมการ 2 มิติได้ดังนี้

$$\nabla^2 G(x,y) = \frac{1}{2\pi\sigma^4}\left(2-\left(\frac{x^2+y^2}{\sigma^2}\right)\right)\cdot\exp\left[-\frac{x^2+y^2}{2\sigma^2}\right] \qquad (3.1)$$

เมื่อ σ คือค่า *space constant* ของ *Gaussian* และนิยามให้ $w = 2\sqrt{2}$ เป็นความกว้างของตำแหน่งศูนย์กลางการกระตุ้นของตัวกระทำการ *(operator)* โดยทั่ว ๆ ไปเราจะกำหนดขนาด s. ของค่าตัวกระทำการที่ประมาณ 3w หรือ 8.5 โดย 99.7 เปอร์เซ็นต์ของพื้นที่ภายใต้ Gaussian จะอยู่ระหว่างบวกและลบสามของค่าเบี่ยงเบนมาตราฐานของค่าเฉลี่ย ซึ่งให้พื้นที่เข้าใกล้ศูนย์มากสำหรับ $\nabla^2 G$ โดยค่าเหล่านี้แสดงไว้ในรูปที่ 3.1

การเปลี่ยนแปลงความเข้มในภาพ I(x,y) สามารถตรวจจับได้โดยการหา *zero-crossing* $\nabla^2 G(x,y) * I(x,y)$ เมื่อ * แทนเครื่องหมาย convolution

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แสดงพารามิเตอร์ของ LoG Masks

Detection of Intensity Changes

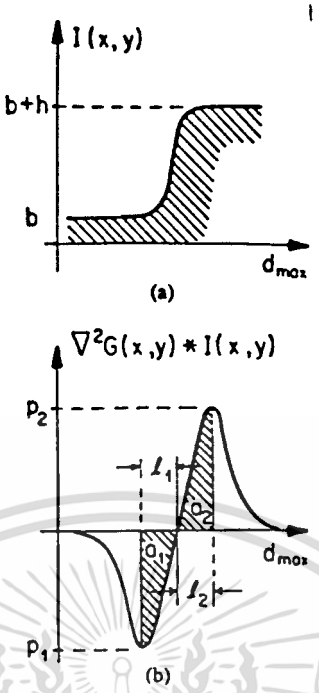
เมื่อใดก็ตามที่มีการเปลี่ยนแปลงความเข้มแสงเกิดขึ้นจะมีทำให้เกิดจุดสูงสุดอันเนื่องมาจาการหาอนุพันธ์อันดับหนึ่ง และจะเกิด *zero-crossing* จากการหาอนุพันธ์อันดับสอง ดังนั้นการตรวจจับการเปลี่ยนความเข้มหลังจากความเข้มแสงที่ลดลง สามารถหาได้จาก *zero-crossing* โดยจะอยู่ในรูปอนุพันธ์อันดับสองของความเข้มแสงในทิศทางที่มีความชันมากที่สุด

ที่จุด *zero-crossing* การใช้ตัวกระทำการที่มีทิศทางอิสระจะทำให้จำนวนความต้องการในการทำ convolution จะลดเป็นหนึ่งและทิศจะได้จากค่าตรงจุดที่มีความชันสูงสุด-และทิศทางที่ได้จะนำมาหาค่าขนาดซึ่งเป็นจะความสัมพันธ์กับแต่ละ zero-crossing

รูปที่ 3. 2 (a) แสดงลักษณะการเปลี่ยนแปลงความเข้มในทิศทาง d_{max} และค่าความชันที่วัดได้ที่ *zero-crossing* แสดงไว้ดังรูปที่ 3. 2(b)

เราจะพิจารณา zero-crossing เป็นจุดที่ตอบสนองของตัวกรอง *(filter)* ที่ผ่านค่าศูนย์ของ 3x3 ในบริเวณใหล้เคียง *(neighborhood)*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดงผลตอบสนองของ LoG masks

เมื่อนำค่า *LoG Mask* มาคำนวณโดยใช้เครื่องคอมพิวเตอร์ที่เป็นดิจิตอลแล้ว ค่าทางสมการคณิตศาสตร์ที่มีลักษระเป็นอนาล็อกจะไม่สามารถคำนวณออกมาได้ จึงต้องมีการแปลงระบบสมการเสียใหม่ โดยแยกสมการ *Gaussian* ออกเป็นสองส่วน แต่ละส่วนถือเป็น 1 มิติอิสระจากกัน นั้นคือแถวและหลักของระบบเมตริกซ์นั่นเอง ซึ่งการทำอย่างนี้แล้วจะทำให้เวลาที่ใช้ในการทำ convolution ลดลงจาก M^2 เป็น 2M ด้วย

ฟังก์ชั่น *LoG* ในสมการที่ 3.1 แยกเป็น *row filter* และ *column filter* ได้ดังนี้

$$\nabla^2 G(x,y) = h_{12}(x,y) + h_{21}(x,y) \qquad (3.2)$$

เมื่อ

$$h_{12}(x,y) = h_1(x)h_2(y) \qquad (3.3)$$

$$h_{21}(x,y) = h_2(x)h_1(y) \qquad (3.4)$$

$$h_1(\xi) = \sqrt{K}\left(1-\frac{\xi^2}{\sigma^2}\right)\exp\left[-\frac{\xi^2}{2\sigma^2}\right] \qquad (3.5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$h_2(\xi) = \sqrt{K}\exp\left[-\frac{\xi^2}{2\sigma^2}\right] \qquad (3.6)$$

ตัวอย่าง พิจารณา *LoG filter* ขนาด 11 x 11 ที่มีค่า σ = 1.41 และ K = 4232 เป็น *row filter* และ *column filter* ขนาด 1 มิติ จากนั้นจึงนำมาสร้างเป็น *LoG filter* 2 มิติ โดยในที่นี้จะแสดงเฉพาะด้านบนซ้ายของ *filter* ด้วยขนาด 1 ใน 4 เท่านั้น ดังนั้นจึงสามารถสร้างเป็น *row* และ *column filter* ได้ดังนี้

$$h_2() = [0 \quad 1 \quad 5 \quad 17 \quad 36 \quad 46 \quad 36 \quad 17 \quad 5 \quad 1 \quad 0]$$
$$h_2() = [-1 \quad -6 \quad -17 \quad -17 \quad 18 \quad 46 \quad 18 \quad -17 \quad -17 \quad -6 \quad -1]$$

นำมาสร้าง *Laplacian - Gaussian filter* (ซึ่งจะแสดงเฉพาะ *upper left quadrant*) ได้ดังนี้

0	−1	−5	−17	−36	−46
−1	−12	−47	−119	−198	−230
−5	−47	−170	−374	−522	−552
−17	−119	−374	−578	−306	0
−36	−198	−522	−306	1296	2484
−46	−230	−522	0	2484	4232

และ *Laplacian-Gaussian filter* (แสดงเฉพาะ *upper left quadrant*) คือ

0	0	0	0	0	0	0	0
0	0	0	0	−1	−2	−3	−4
0	0	0	−1	−6	−19	−35	−43
0	0	−1	−10	−43	−114	−196	−233
0	−1	−6	−43	−165	−369	−521	−558
0	−2	−19	−114	−369	−573	−303	0
0	−3	−35	−196	−521	−303	1283	2472
0	−4	−43	−233	−558	0	2472	4232

อัลกอริทึมพื้นฐานในการกำหนดตำแหน่งขอบและทิศทางของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อใช้ *Second Derivative Edge Operator* เช่น *LoG* แล้วขอบจะถูกจับได้ โดยดูจากเอาท์พุทของการทำ convolution ที่มีการเปลี่ยนเครื่องหมายเป็นตรงข้าม โดย มี *Zero-crossing* อยู่ตรงกลาง เช่น ในระบบหนึ่งมิติที่มีจุดภาพ 3 จุดเรียงต่อกันในแนวนอนแล้วจะได้ว่าจุดตรงกลางจะต้องมีค่าเป็นศูนย์ในขณะที่อีกสองจุดรอบ ๆ ข้างจะต้องมีเครื่องหมายตรง ข้ามกัน

สำหรับในระบบ 2 มิติ จะต้องกำหนดให้จุดกึ่งกลางล้อมรอบด้วยเมตริกซ์ขนาด 3 x 3 เสียก่อน จากนั้นนำค่าที่ได้จากการทำ *convolution* ไปตรวจสอบกับรูปแบบมาตราฐานที่กำหนดขึ้นเพื่อหาทิศทาง โดยรูปแบบทิศทางทั้งหมดจะแสดงไว้ในภาคผนวก ข.

อัลกอริทึมในการหาจุดที่เหมือนกัน

สำหรับเนื้อหาข้างต้นเป็นทฤษฎี *Zero crossing* ซึ่งเมื่อนำวิธีข้างต้นไปกระทำการกับแต่ละจุดของภาพแล้วให้ออกมาเป็น

1.) ขนาดที่ได้จากการกระทำการ
2.) เครื่องหมายที่ได้จากการกระทำการ
3.) ทิศทางที่ได้จากการกระทำการ

สรุปได้ว่าภาพที่ผ่านกระบวนการ *Zero crossing* แล้วแต่ละจุดของภาพจะถูกเปลี่ยนเป็นข้อมูลทั้ง 3 ข้างบนนั้นเอง

ดังนั้นอัลกอริทึมของการหาจุดที่เหมือนกันของภาพสองภาพจะเป็นดังนี้

1.) นำภาพที่ถ่ายจากกล้องตัวที่ 1 มาผ่านกระบวนการ Zero crossing โดยเก็บข้อมูลที่ได้ไว้ในไฟล์ data1.doc

2.) นำภาพที่ถ่ายจากกล้องตัวที่ 2 มาผ่านกระบวนการ Zero crossing โดยเก็บข้อมูลที่ได้ไว้ในไฟล์ data2.doc

3.) นำข้อมูลที่ได้จากทั้ง 2 ไฟล์มาทำการเปรียบเทียบกัน โดยจะต้องเปรียบเทียบแบบจุดต่อจุดจนครบทุกจุด ซึ่งจะต้องมีจุดอยู่คู่หนึ่งที่เป็นจุดที่เหมือนกันพอดี โดยเราจะตั้งสมมุติฐานว่าจุดที่เหมือนกันจะต้องเป็นไปตามเงื่อนไขต่อไปนี้

- มีเครื่องหมายเหมือนกัน
- มีทิศทางเหมือนกัน
- ผลต่างของขนาดของจุดทั้งสองจุดจะต้องมีค่าน้อยที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

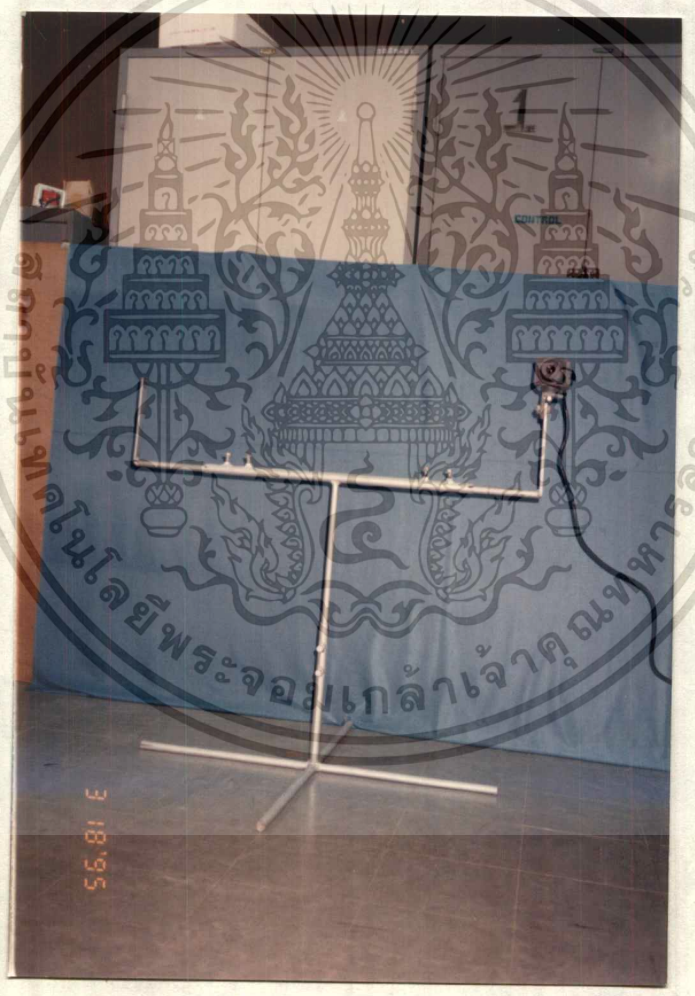
บทที่ 4

การทดลองหาระยะทาง

สำหรับเนื้อหาในบทนี้จะเป็นการกล่าวถึงการทดลองที่ใช้ทดสอบทฤษฎีต่าง ๆ ที่กล่าวมาแล้วว่าจะสามารถนำไปใช้งานจริงได้เพียงใด

ขั้นตอนการทดลอง สามารถแยกเป็นข้อ ๆ ได้ดังนี้

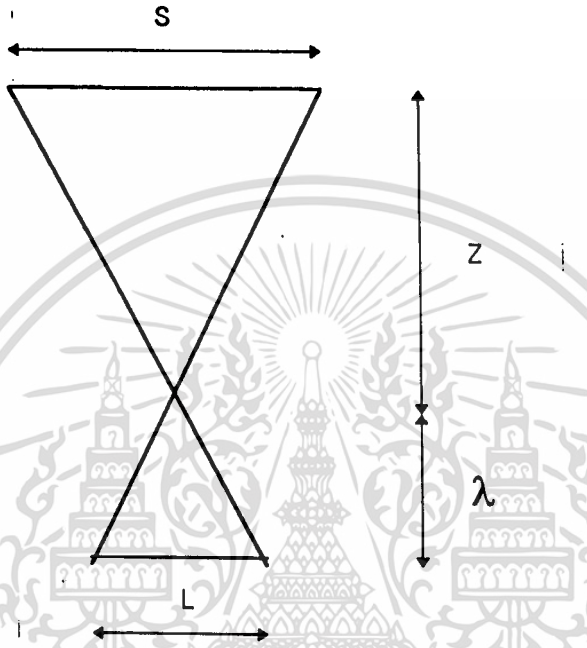
1.) จัดทำขาตั้งกล้องดังรูปที่ 4.1 โดยขาตั้งกล้องที่ทำขึ้นนี้จะใช้ในการวางกล้องตามระบบพิกัดที่เรากำหนดขึ้น



รูปที่ 4.1 ขาตั้งและกล้องที่ใช้ในการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.) คำนวณหาค่า K ซึ่งเป็นค่าคงที่ที่ใช้ในการแปลงจำนวนจุดเป็นความยาวเซ็นติเมตร โดยจะต้องนำวัตถุและกล้องวางดังรูปที่ 4.2 สำหรับวัตถุที่ใช้ในที่นี้คือกล่องกระดาษสีดำที่รู้ความกว้างแน่นอน (ความกว้างของกล่องเท่ากับ 23.5 เซ็นติเมตร) จากนั้นจึงเปลี่ยนระยะห่างระหว่างวัตถุกับกล้องไปเรื่อย ๆ (ดังตารางผลการทดลองที่ 4.1) พร้อมกับบันทึมค่าระยะห่างระหว่างวัตถุกับกล้อง และจำนวนความกว้างของวัตถุ (วัดเป็นจำนวนจุด) เมื่อมีการเปลี่ยนระยะห่างระหว่างวัตถุกับกล้อง



รูปที่ 4.2 แสดงการเซ็ตระบบเพื่อใช้หาค่า K

เมื่อ S = ความกว้างของวัตถุที่ใช้ถ่าย (เซนติเมตร)

Z = ระยะทางระหว่างกล้องกับวัตถุ (เซนติเมตร)

λ = ระยะโฟกัสของกล้อง ซึ่งกล้องที่ใช้ในที่นี้มีโฟกัส 0.28 เซนติเมตร

L = ความกว้างของวัตถุที่วัดได้จากภาพ มีหน่วยเป็นจำนวนจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 การหาค่า K

ครั้งที่	Z (cm.)	L (pixels)	K (x10 $^{-4}$ cm./pixel)
1	114	113	5.11
2	183	67	5.36
3	91	134	5.396
4	137	89	5.396
5	45.5	274	5.278
6	202	61	5.308
7	177	71	5.2
8	124	101	5.251
9	60.5	205	5.3
10	71.8	171	5.359

$$\overline{K} = \frac{5.11+5.36+5.396+5.396+5.278+5.308+5.2+5.251+5.3+5.359}{10} x10^{-4} = 5.296x10^{-4}$$

จากการทดลองจะสามารถหาค่า K เฉลี่ยได้เท่ากับ 5.296 x 10^{-4} cm. / pixel

3.) นำกล้อง CCD สองตัวไปวางไว้ที่จุดวางกล้องบนขาตั้งกล้อง โดยจะวางไว้คนละด้านของขาตั้งกล้อง แต่เนื่องจากเรามีกล้อง CCD เพียงตัวเดียว ดังนั้นในการทดลองจริงจึงใช้กล้องตัวเดียวนี้แต่เปลี่ยนการถ่ายภาพไปทีละด้าน โดยกำหนดให้กล้องที่วางอยู่ดังรูปที่ 4.1 เป็นกล้องตัวที่ 1 และเมื่อเปลี่ยนกล้องมาวางอีกด้านจะกำหนดให้เป็นกล้องตัวที่ 2

4.) เก็บภาพของวัตถุโดยใช้การ์ดดิจิไตเซอร์ในการแปลงสัญญาณภาพเป็นสัญญาณดิจิตอลซึ่งจะทำให้ได้ภาพขนาด 512 x 256 จุด ดังรูปที่ 4.3 การเก็บภาพจะต้องเก็บ 2 รูปคือภาพที่เก็บจากกล้องตัวที่ 1 (กล้องด้านขวา) และภาพที่เก็บจากกล้องตัวที่ 2 (กล้องด้านซ้าย) เมื่อได้ภาพที่เก็บจากกล้องตัวที่ 1 แล้วจะเก็บภาพนั้นในแฟ้มข้อมูลชื่อ *data1.dat* และภาพที่ได้จากกล้องตัวที่ 2 จะเก็บในแฟ้มข้อมูลชื่อ *data2.dat*

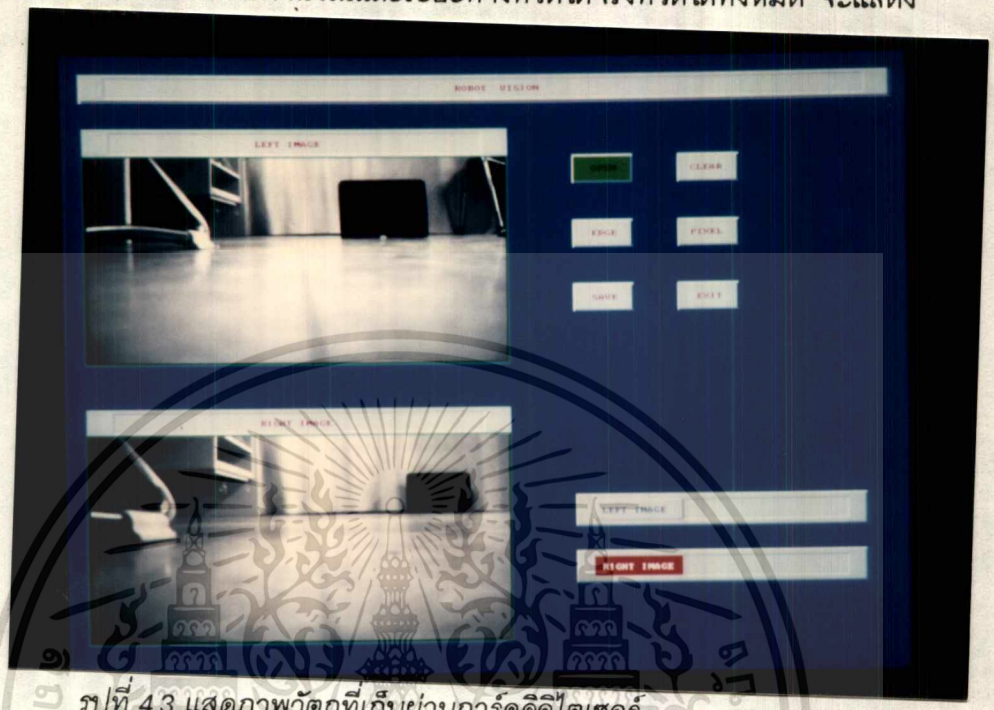
5.) ทำการทดลองหาระยะห่างระหว่างวัตถุกับกล้องโดยเปลี่ยนตำแหน่งของวัตถุไปเรื่อย ๆ (ดังตารางผลการทดลองที่ 4.2) วัตถุที่ใช้จะเป็นกล่องกระดาษสีดำ พร้อมกับเก็บข้อมูลภาพที่ตำแหน่งวัตถุต่าง ๆ กัน

5.) นำภาพที่ได้จากกล้องตัวที่ 1 และ 2 ของวัตถุเดียวกันที่ระยะห่างเดียวกันมาหา *Zero - crossing* ดังรูปที่ 4.4 พร้อมกับเก็บข้อมูลภาพที่ได้จากกล้องตัวที่ 1 ที่ผ่านการ *Zero - crossing* แล้วไว้ในแฟ้มข้อมูล *zero1.dat* และภาพจากกล้องตัวที่ 2 ที่ผ่านการทำ *Zero - crossing* แล้วไว้ในแฟ้มข้อมูล *zero2.dat*

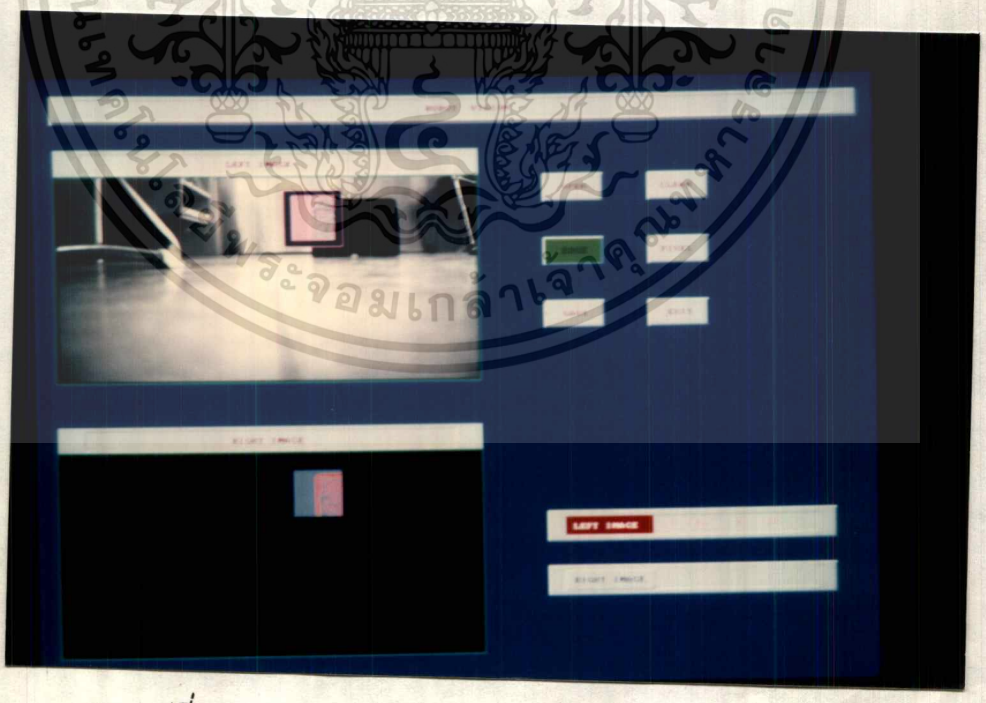
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.) นำข้อมูลจากแฟ้มข้อมูล *zero1.dat* และ *zero2.dat* มาเปรียบเทียบหาจุดที่เหมือนกัน ซึ่งจะทำให้ได้คู่ลำดับของภาพที่ 1 คือ (x_1 , y_1) และของภาพที่ 2 คือ (x_2 , y_2)

7.) นำคู่ลำดับที่ได้ไปคำนวณหาระยะทางต่อไป (โดยใช้สมการที่ 2.15 , 2.16 , 2.17 , 2.18 และ 2.19 ตามลำดับ) ซึ่งผลการคำนวณและระยะทางที่วัดได้จริงที่วัดได้ทั้งหมด จะแสดงอยู่ในตารางที่ 4.2



รูปที่ 4.3 แสดภาพวัตถุที่เก็บผ่านการ์ดดิจิไตเซอร์



รูปที่ 4.4 แสดงการทำ Zero crossing กับภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 ตารางผลการทดลอง

กรณีที่ 1 เมื่อ $Y_0 = 0$ และ $Z_0 = 0$

ครั้งที่	1	2	3	4
X_0 (cm.)	20.5	25.5	30.5	35.5
X (ที่วัดจริง เป็น cm.)	13.7	12.5	11.2	37.3
Y (ที่วัดจริง เป็น cm .)	293.5	100.1	133.7	160.7
Z(ที่วัดจริง เป็น cm.)	25.4	0	25.4	25.4
d (ที่วัดจริง เป็น cm.)	293.5	100.1	133.7	160.7
x_1 (จุด)	-20	-100	-80	-11
y_1 (จุด)	15	2	50	70
x_2 (จุด)	57	206	154	201
y_2 (จุด)	21	7	59	65
X(จากการคำนวณเป็น cm.)	9.57	8.55	9.37	31.54
Y (จากการคำนวณเป็น cm.)	281.79	88.39	138.1	177.34
Z(จากการคำนวณเป็น cm.)	28.52	-0.33	-21.94	-23.44
d (จากการคำนวณเป็น cm.)	283.44	90.1	141.42	178.93
%ความผิดพลาด	3.43	10.01	5.78	10.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 2 เมื่อ $Y_0 = 0$ และ $Z_0 = 77$ cm.

ครั้งที่	1	2	3	4
X_0 (cm.)	20.5	25.5	30.5	35.5
X (ที่วัดจริง เป็น cm.)	13.8	9.1	13.2	30.4
Y (ที่วัดจริง เป็น cm .)	212.8	100.2	175.2	205.1
Z(ที่วัดจริง เป็น cm.)	62.6	69.2	58.5	75.2
d (ที่วัดจริง เป็น cm.)	213.39	101.83	177.02	205.17
x_1 (จุด)	-26	-150	-63	-25
y_1 (จุด)	3	14	55	12
x_2 (จุด)	63	163	144	182
y_2 (จุด)	10	8	43	15
X(จากการคำนวณเป็น cm.)	8.24	0.77	11.65	26.65
Y (จากการคำนวณเป็น cm.)	243.8	76.83	156.08	181.62
Z(จากการคำนวณเป็น cm.)	75.62	79.28	60.79	72.88
d (จากการคำนวณเป็น cm.)	244.11	80.74	158.04	181.88
%ความผิดพลาด	14.39	20.71	10.72	11.35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

จากการทดลองพบว่าจะเกิดค่าความผิดพลาดขึ้นค่าหนึ่ง ซึ่งสามารถคิดเป็นค่าเฉลี่ยได้ดังนี้

$$\%\text{ความผิดพลาดเฉลี่ย} = \frac{10.1+5.78+3.43+10.18+14.39+20.71+10.72+11.35}{8} = 10.83$$

ซึ่งถ้าการเซ็ตระบบเป็นไปตามการทดลองกรณีที่ 1 ($Y_0 = 0$ และ $Z_0 = 0$) หรือการวางกล้องไว้บนพื้นจะให้เปอร์เซ็นต์ความผิดพลาดน้อยกว่ากรณีที่ 2 ($Y_0 = 0$ และ $Z_0 = 77$ cm.) พอสมควร

ข้อจำกัดของการวัดระยะทางโดยวิธีนี้ก็คือ

- วัตถุที่จะวัดระยะทางจะต้องมีจุดที่เหมือนกันในภาพทั้งสองที่ถ่ายได้
- วัตถุที่จะวัดระยะทางจะต้องเป็นวัตถุที่มีรูปทรงทางเรขาคณิตอย่างง่าย ๆ เนื่องจากถ้าเป็นรูปทรงที่ซับซ้อนแล้ว วิธี Zero crossing จะไม่สามารถหาจุดที่เหมือนกันได้อย่างถูกต้อง

วิจารณ์ผลการทดลอง

จากการทดลองพบว่ามีค่าความผิดพลาดเกิดขึ้นประมาณ 10 % โดยค่าความผิดพลาดที่เกิดขึ้นนี้อาจเกิดจาก

- ระบบพิกัดและสมการหาระยะทางที่คิดขึ้นมาเป็นจริงในทางอุดมคติ แต่ในทางปฏิบัติจะพบข้อผิดพลาดพอสมควร ดังนั้นควรเปลี่ยนระบบพิกัดและสมการใหม่
- อัลกอริทึมที่ใช้ในการหาจุดที่เหมือนกันมีข้อผิดพลาด โดยส่วนหนึ่งเป็นผลมาจากความผิดพลาดของตัวอัลกอริทึมเองและอาจเกิดความผิดพลาดของโปรแกรมที่เขียนขึ้น
- การ์ดแปลงสัญญาณจากกล้องมาเข้าเครื่องคอมพิวเตอร์มีสัญญาณรบกวนมาก
- เครื่องมือที่ใช้ในการวัดไม่เที่ยงตรงโดยเฉพาะการวัดที่ต้องการความตั้งฉากหรือวัดมุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1.) Andres Huertas and Gerard Medioni , " DETECTION OF INTENSITY CHANGES WITH SUBPIXEL ACCURACY USING LAPLACIAN-GAUSSIAN MASKS " , IEEE Trans. Pattern Anal . Machine Intell . , VOL.PAMI-8NO.5 Sep 1986 .

2.) Adrian Low , " INTRODUCTORY COMPUTER VISION AND IMAGE PROCESSING " , MaGRAW -HILL Book Company , 1991 .

3.) Rafael C. Conzales , " DIGITAL IMAGE PROCESSING " , Second Edition , Addision-Wesley Puslishing Company , 1987 .

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

ไฟล์ mo.c

```
/*********************************************************/
/*    Program read pioture (512 x 256 pixcels )          */
/*       Version 1.1 ( Graphics interface )              */
/*       Product by Engineer of Ladkrabang               */
/*           Borlande++ Complier Version 3.1             */
/*    This version use getpixel in thinning function     */
/*********************************************************/
#include<stdio.h>
#include<conio.h>
#include<graphics.h>
#include<alloc.h>
#include<key.h>
#include<dos.h>
#include<new.h>
#include<string.h>
#include<bios.h>
#include<process.h>
#include<math.h>
#include<stdlib.h>

#define MAXC 512
#define MAXR 256
#define NOTFOUND 5
#define ENTER 0
#define NOT_ENTER 1
#define SHOWX 24
#define SHOW1Y 120
#define SHOW2Y 470

void zerocrossing(int sigma,int k,int rando,int x1,
                  int y1,int x2,int y2,int winselect);
int comparedirect(long int z[],int winselect,int x,int y);
void opengraph(void),bottom(int x,int y,int s,char *ch),filename(void);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void show(int x,int y,char *ch);
void createscreen(void);
int huge detect256(void);
int getvpage(void);
void spacetoselect(int mo,int om,int enter);
void clearwin(int x1,int y1,int x2,int y2);
void drawpicture(char iname[50],int winselect);
void ver_edge_sobel(char iname[50],int winselect,int threshold);
void hor_edge_sobel(int winselect,int threshold);
void barselectwin(int x,int y,int test1,int sw1,char *ch);
void cursor(int row,int col);
void filenotfound(void);
void selectpixel(int winselect);
void save(int winselect);
void clear(int winselect);
int histrogram(char iname[50],int winselect);
//int thinning(int winselect,int method);
int compare(void);
int readkey(void);
void knit_song(void);
void showtitle(void);
void create_border(int fx,int fy,int lx,int ly,int color);
void displayvalue(int x,int y,int winselect);
void savepicture(char iname[50],int winselect);
void selectboundary(int winselect);
void crosssign(int x,int y);
void matching(void);

void main(void)
{
  FILE *ifp;
  char iname[50],color,*ch;
  unsigned key = 0x0000;
  int y,i,pel,level,maxx,maxy,x,winselect=1;
  int testfound,method,space1 = 1,space2 = 1,threshold,result;
  opengraph();     // begin graphics
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
oreatesoreen();  // draw soreen
showtitle();
kmit_song();
spaoetosoleot(spaoe1,1,NOT_ENTER);  //  OPEN bottom
do
 {
      if(key == F1_Key)
       {
         stropy(oh,"LEFT IMAGE");
         barsoleotwin(620,550,0,1,oh);  // 1st barsoleotwin
         stropy(oh,"RIGHT IMAGE");
         barsoleotwin(620,620,0,0,oh);  // 2nd barsoleotwin
         winsoleot = 1;
       }
      else if(key == F2_Key)
       {
         stropy(oh,"RIGHT IMAGE");
         barsoleotwin(620,620,0,1,oh);  // 2nd barsoleotwin
         stropy(oh,"LEFT IMAGE");
         barsoleotwin(620,550,0,0,oh);  // 1st barsoleotwin
         winsoleot = 2;
       }
      else if(key == Alt_O)
       {
         spaoetosoleot(spaoe1,0,NOT_ENTER);
         spaoe1 = 1;
         spaoetosoleot(spaoe1,1,ENTER);
         filename();
         oursor(27,87);
         soanf("%s",inamo);
         x = 600 ; y = 407;
         setfillstyle(1,1);
         bar(x-2,y-2,x+352,y+102);
         drawpioture(inamo,winsoleot);
         spaoetosoleot(spaoe1,1,NOT_ENTER);
       }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else if(key == Alt_C) // Clear window
  {
    spacetoselect(space1,0,NOT_ENTER);
    space1 = 2;
    spacetoselect(space1,1,ENTER);
    clear(winselect);
    spacetoselect(space1,1,NOT_ENTER);
  }
else if(key == Alt_D)
  {
    spacetoselect(space1,0,NOT_ENTER);
    space1 = 3;
    spacetoselect(space1,1,ENTER);
    selectboundary(winselect);
    spacetoselect(space1,1,NOT_ENTER);
  }
else if(key == Alt_P)  // Pixel
 {
    spacetoselect(space1,0,NOT_ENTER);
    space1 = 4;
    spacetoselect(space1,1,ENTER);
    selectpixel(winselect);
    spacetoselect(space1,1,NOT_ENTER);
 }
else if(key == Alt_S)
 {
    spacetoselect(space1,0,NOT_ENTER);
    space1 = 5;
    spacetoselect(space1,1,ENTER);
    matching();
    spacetoselect(space1,1,NOT_ENTER);
 }
key = bioskey(0);
if(key == Enter_Key)
 {
  if(space1 == 1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        key = Alt_O;
      else if(space1 == 2)
        key = Alt_C;
      else if(space1 == 3)
        key = Alt_D;
      else if(space1 == 4)
        key = Alt_P;
      else if(space1 == 5)
        key = Alt_S;
      else if(space1 == 6)
        key = Alt_X;
    }
  else if(key == Left_Key)
    {
      if((space1 != 1) && (space1 != 3) && (space1 != 5))
        {
          space2 = space1;
          space1 = space1-1;
          spacetoselect(space2,0,NOT_ENTER);
          spacetoselect(space1,1,NOT_ENTER);
        }
    }
  else if(key == Right_Key)
    {
      if((space1 != 2) && (space1 != 4) && (space1 != 6))
        {
          space2 = space1;
          space1 = space1+1;
          spacetoselect(space2,0,NOT_ENTER);
          spacetoselect(space1,1,NOT_ENTER);
        }
    }
  else if(key == Up_Key)
    {
      space2 = space1;
      if((space1 == 1) || (space1 == 2))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                space1 = space1;
            else
                space1 = space1-2;
            spacetoselect(space2,0,NOT_ENTER);
            spacetoselect(space1,1,NOT_ENTER);
          }
        else if(key == Down_Key)
          {
            space2 = space1;
            if((space1 == 6) || (space1 == 5))
                space1 = space1;
            else
                space1 = space1+2;
            spacetoselect(space2,0,NOT_ENTER);
            spacetoselect(space1,1,NOT_ENTER);
          }
    }
  while(key != Alt_X);
  spacetoselect(space2,0,NOT_ENTER);
  spacetoselect(space1,1,ENTER);
  delay(500);
  spacetoselect(space1,1,NOT_ENTER);
  delay(500);
  closegraph();
}

int huge detect256(void)
{
  return 4;
  /* 0 =  320 * 200
     1 =  640 * 350
     2 =  640 * 480
     3 =  800 * 600
     4 = 1024 * 768
  */
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void opengraph(void)
{
  int gd,gm,y,i;
  installuserdriver("svga256",detect256);
  gd = DETECT;
  initgraph(&gd,&gm,"d:\borlandc\bin");
  y=0;
  for(i=192;i<256;i++)
     {
          setrgbpalette(i,y,y,y);
          y++;
     }
}

void createscreen(void)
{
    char *ch;
    int x,y,maxx,maxy;
    maxx = getmaxx();
    maxy = getmaxy();
    setfillstyle(1,1);
    bar(0, 0,maxx, maxy);      // fill background with blue color
    setfillstyle(1,80);
    bar(22,22,maxx-22,50);
    setfillstyle(1,24);
    bar(22,51,maxx-20,54);
    bar(maxx-20,22,maxx-19,54);
    setfillstyle(1,82);
    bar(19,20,maxx-22,22);
    bar(19,20,22,54);          // draw title bar
    setcolor(24);
    line(50,24,maxx-50,24);
    line(50,24,50,48);
    setcolor(15);
    line(50,48,maxx-50,48);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    line(maxx-50,48,maxx-50,24);
    setcolor(4);
    outtextxy((maxx/2)-35,33,"ROBOT  VISION"); // write "ROBOT VISION" tille*/

    strcpy(ch,"LEFT IMAGE");
    show(SHOWX,SHOW1Y,ch);
    strcpy(ch,"RIGHT IMAGE");
    show(SHOWX,SHOW2Y,ch);     // prepare 1st window and 2nd window

    strcpy(ch," OPEN");
    bottom(620,120,1,ch);  // bottom OPEN
    strcpy(ch,"CLEAR");
    bottom(750,120,1,ch);  // bottom SAVE
    strcpy(ch," EDGE");
    bottom(620,200,1,ch);  // bottom EDGE
    strcpy(ch,"PIXEL");
    bottom(750,200,1,ch);   // bottom EXIT
    strcpy(ch," SAVE");
    bottom(620,280,1,ch);  //  bottom SAVE
    strcpy(ch," EXIT");
    bottom(749,280,1,ch);  // bottom TOOL

    strcpy(ch,"LEFT IMAGE");
    barselcotwin(620,550,1,1,ch);
    strcpy(ch,"RIGHT IMAGE");
    barselcotwin(620,620,1.0,ch); // draw 1st barselcot and 2nd barselcot
}

void drawpicture(char iname[50],int winselcot)
{
    int line_y,pel,i,wx = SHOWX,wy;
    FILE *ifp;
    unsigned char *temp = new unsigned char[MAXC];
    if((ifp=fopen(iname,"r+b"))==NULL)
      {
          filenotfound();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            folose(ifp);
        }
    else
        {
            if(winselcot == 1)
                wy = SHOW1Y;
            else
                wy = SHOW2Y;
            for(linc_y=0;linc_y<MAXR;linc_y++)
              {
                fread(temp,sizeof(unsigned char),MAXC,ifp);
                for(pcl=0;pcl<MAXC;pcl++)
                    {
                    if(temp[pcl] > 63)
                      {
                        temp[pcl] = 63;
                        putpixel(pcl+wx,linc_y+wy,temp[pcl]+192);
                      }
                    else
                        putpixel(pcl+wx,linc_y+wy,temp[pcl]+192);
                    }
                }
              delete temp;
              folose(ifp);
        }
}

void save(int winselcot)
{
    int pcl,linc_y,wx=SHOWX,wy;
    FILE *fp;
    if((fp=fopen("ts.tmp","w+b"))== NULL)
            exit(1);
    if(winselcot == 1)
      wy = SHOW1Y;
    else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    wy = SHOW2Y;
  for(pel=0;pel<MAXC;pel++)
   {
       for(line_y=0;line_y<MAXR;line_y++)
        {
          unsigned char value = getpixel(wx+MAXC-1-pel,wy+line_y)-192;
          fwrite(&value,sizeof(unsigned char),1,fp);
        }
   }
  fclose(fp);
}

void ver_edge_sobel(char iname[50],int winselect,int threshold)
{
  int line_y,pel,i,m,wx = SHOWX,wy,deltax,deltay,x,y;
  FILE *ifp;
  unsigned char *temp = new unsigned char[MAXC];
  unsigned char *r1 = new unsigned char[MAXC];
  unsigned char *r2 = new unsigned char[MAXC];
  unsigned char value;
  if((ifp=fopen(iname,"r+b"))==NULL)
   {
    filenotfound();
    fclose(ifp);
   }
  else
   {
       if(winselect == 1)
            wy = SHOW1Y;
       else
            wy = SHOW2Y;
       for(line_y=0;line_y<MAXR;line_y++)
        {
            fread(temp,sizeof(unsigned char),MAXC,ifp);
            for(pel=0;pel<MAXC;pel++)
            {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if(linc_y == 0)
         {
    r1[pel] = temp[pel];
      putpixel(wx+pel,wy+linc_y,192);
         }
        else if(linc_y == 1)
         {
          r2[pel] = temp[pel];
          if((pel == 0) || (pel == MAXC-1))
          putpixel(wx+pel,wy+linc_y,192);
         }
        else  // linc_y > 1
        {
          if((pel == 0) || (pel == MAXC-1))
              {
                putpixel(wx+pel,wy+linc_y,192);
              }
          else
            {
              deltax = (r1[pel-1]+2*r2[pel-1]+temp[pel-1]) -
                       (r1[pel+1]+2*r2[pel+1]+temp[pel+1]);
              deltay = (temp[pel-1]+2*temp[pel]+temp[pel+1]) -
                       (r1[pel-1]+2*r1[pel]+r1[pel+1]);
              m = sqrt(pow(deltax,2) + pow(deltay,2));
              if(m < threshold)
                {
                    putpixel(wx+pel,wy+linc_y-1,192);
                }
              else if(m >= threshold)
                {
                    putpixel(wx+pel,wy+linc_y-1,192+63);
                }
            }
        }
      }
    r1 = r2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                r2 = temp;
            }
          deltax = (r1[pel-1]+2*r2[pel-1]+temp[pel-1]) -
                   (r1[pel+1]+2*r2[pel+1]+temp[pel+1]);
          deltay = (temp[pel-1]+2*temp[pel]+temp[pel+1]) -
                   (r1[pel-1]+2*r1[pel]+r1[pel+1]);
          m = sqrt(pow(deltax,2) + pow(deltay,2));
          if(m >= threshold)
            {
              putpixel(wx+pel,wy+MAXR-2,63+192);
            }
          else if(m < threshold)
            {
              putpixel(wx+pel,wy+MAXR-2,192);
            }
          fclose(ifp);
          delete r1;
          delete r2;
          delete temp;
    }
}

void hor_edge_sobel(int winselect,int threshold)
{
  int line_y,pel,i,m1,m2,wx = SHOWX,wy,deltax,deltay,x,y;
  FILE *ifp;
  unsigned char *temp = new unsigned char[MAXR];
  unsigned char *r1 = new unsigned char[MAXR];
  unsigned char *r2 = new unsigned char[MAXR];
  unsigned char value;
  if((ifp=fopen("ts.tmp","r+b"))==NULL)
   {
    filenotfound();
    fclose(ifp);
   }
  else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
    if(winselect == 1)
        wy = SHOW1Y;
    else
        wy = SHOW2Y;
    for(pel=0;pel<MAXC;pel++)
      {
        fread(temp,sizeof(unsigned char),MAXR,ifp);
        for(line_y=0;line_y<MAXR;line_y++)
         {
           if(pel == 0)
            {
                r1[line_y] = temp[line_y];
            }
           else if(line_y == 1)
           {
            r2[line_y] = temp[line_y];
            if((line_y == 0) || (line_y == MAXR-1))
              {
              }
           }
          else  // line_y > 1
          {
           if((line_y == 0) || (line_y == MAXC-1))
                {
                }
           else
                {
                  deltax = (r1[line_y-1]+2*r2[line_y-1]+temp[line_y-1]) -
                           (r1[line_y+1]+2*r2[line_y+1]+temp[line_y+1]);
                  deltay = (temp[line_y-1]+2*temp[line_y]+temp[line_y+1]) -
                           (r1[line_y-1]+2*r1[line_y]+r1[line_y+1]);
                  m1 = sqrt(pow(deltax,2) + pow(deltay,2));
                  if(getpixel(wx+MAXC-1-pel,wy+line_y-1) == 63+192)
                    {
                      putpixel(wx+MAXC-1-pel,wy+line_y-1,192+63);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    }
                    else
                    {
                        if(m1 < threshold)
                            putpixel(wx+MAXC-1-pcl,wy+linc_y-1,192);
                        else if(m1 >= threshold)
                            putpixel(wx+MAXC-1-pcl,wy+linc_y-1,192+63);
                    }
                }
            }
        }
        r1 = r2;
        r2 = temp;
    }
    deltax = (r1[linc_y-1]+2*r2[linc_y-1]+temp[linc_y-1]) -
             (r1[linc_y+1]+2*r2[linc_y+1]+temp[linc_y+1]);
    deltay = (temp[linc_y-1]+2*temp[linc_y]+temp[linc_y+1]) -
             (r1[linc_y-1]+2*r1[linc_y]+r1[linc_y+1]);
    m1 = sqrt(pow(deltax,2) + pow(deltay,2));
    if(getpixel(wx+MAXC-1-pcl,wy+linc_y-1) == 63+192)
    {
        putpixel(wx+MAXC-1-pcl,wy+linc_y-1,192+63);
    }
    else
    {
        if(m1 < threshold)
            putpixel(wx+MAXC-1-pcl,wy+linc_y-1,192);
        else if(m1 >= threshold)
            putpixel(wx+MAXC-1-pcl,wy+linc_y-1,192+63);
    }
    fclose(ifp);
    delete r1;
    delete r2;
    delete temp;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void show(int x,int y,char *ch)
{
    setfillstyle(1,80);
    bar(x-2,y-35,x+MAXC+2,y-2);  // fill background for barselect
    setfillstyle(1,2);
    bar(x-2,y-2,x+MAXC+2,y);
    bar(x-2,y-2,x,y+MAXR+2);
    bar(x-2,y+MAXR,x+MAXC+2,y+MAXR+2);
    bar(x+MAXC+1,y-2,x+MAXC+2,y+MAXR+2);
    setcolor(24);
    line(x+30,y-30,x+MAXC-30,y-30);
    line(x+30,y-30,x+30,y-5);
    setcolor(15);
    line(x+30,y-5,x+MAXC-30,y-5);
    line(x+MAXC-30,y-5,x+MAXC-30,y-30);
    setfillstyle(1,0);
    bar(x,y,x+MAXC,y+MAXR);
    setcolor(4);
    outtextxy(x+210,y-20,ch);  // show title barselect name
}

void bottom(int x,int y,int select,char *ch)
{
    setfillstyle(1,80);
    bar(x,y,x+70,y+30);
    if(select == 1)
     {
          setfillstyle(1,24);
          bar(x,y+31,x+72,y+32);
          bar(x+71,y+31,x+72,y+32);
          setfillstyle(1,15);
          bar(x-2,y-2,x-1,y+32);
          bar(x-2,y-2,x+72,y-1);
     }
    else    // select == 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
        setfillstyle(1,15);
        bar(x,y+31,x+72,y+32);
        bar(x+71,y+31,x+72,y+32);
        setfillstyle(1,24);
        bar(x-2,y-2,x-1,y+32);
        bar(x-2,y-2,x+72,y-1);
    }
  setcolor(4);
  outtextxy(x+14,y+12,ch);
}

void barselectwin(int x,int y,int test1,int sw1,char *ch)
{
  if(test1 == 1)
    {
        setfillstyle(1,80);
        bar(x,y,x+350,y+35);
        setfillstyle(1,24);
        bar(x,y+36,x+352,y+37);
        bar(x+351,y+36,x+352,y);
        setfillstyle(1,15);
        bar(x-2,y-2,x-1,y+36);
        bar(x-2,y-2,x+352,y-1);
        setcolor(15);
        line(x+130,y+30,x+330,y+30);
        line(x+330,y+5,x+330,y+30);
        setcolor(24);
        line(x+130,y+5,x+130,y+30);
        line(x+130,y+5,x+330,y+5);
    }
    if(sw1 == 1)  // select window 1
    {
        setfillstyle(1,4);
        bar(x+20,y+5,x+129,y+30);
        setcolor(15);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            line(x+20,y+30,x+129,y+30);
            line(x+129,y+30,x+129,y+5);
            outtextxy(x+30,y+15,oh);
            setcolor(24);
            line(x+20,y+5,x+20,y+30);
            line(x+20,y+5,x+129,y+5);
        }
    else  // don ' t select window 1
        {
            setfillstyle(1,80);
            bar(x+20,y+5,x+129,y+30);
            setcolor(24);
            line(x+20,y+30,x+129,y+30);
            line(x+129,y+30,x+129,y+5);
            setcolor(8);
            outtextxy(x+30,y+15,oh);
            setcolor(15);
            line(x+20,y+5,x+20,y+30);
            line(x+20,y+5,x+129,y+5);
        }
}

void filename(void)
{
    int x = 600 ,y = 407;
    setfillstyle(1,80);
    bar(x,y,x+350,y+100);
    setfillstyle(1,24);
    bar(x,y+101,x+352,y+102);
    bar(x+351,y+101,x+352,y);
    setfillstyle(1,82);
    bar(x-2,y-2,x-1,y+101);
    bar(x-2,y-2,x+352,y-1);
    setcolor(15);
    line(x+85,y+30,x+330,y+30);
    line(x+330,y+5,x+330,y+30);
```

เอกสารนี้มีไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    setcolor(24);
    line(x+85,y+5,x+85,y+30);
    line(x+85,y+5,x+330,y+5);
    setfillstyle(1,0);
    bar(x+86,y+6,x+329,y+29);
    setcolor(14);
    outtextxy(x+10,y+15,"FILENAME");
}

void spaceto select(int method,int select,int enter)
{
  int om,x[7],y[7];
  x[1] = 620; y[1] = 120; // open
  x[2] = 750; y[2] = 120; // save
  x[3] = 620; y[3] = 200; // edge
  x[4] = 750; y[4] = 200; // exit
  x[5] = 620; y[5] = 280; // mapping
  x[6] = 750; y[6] = 280; // tool
       if(select == 0)  /* not select */
         {
            setfillstyle(1,80);
            bar(x[method],y[method],x[method]+70,y[method]+30);
            setcolor(4);
         }
       else  // select
         {
            setfillstyle(1,2);
            bar(x[method],y[method],x[method]+70,y[method]+30);
            setcolor(0);
         }

       if(method == 1)
         outtextxy(x[method]+14,y[method]+12," OPEN");
       else if(method == 2)
         outtextxy(x[method]+14,y[method]+12,"CLEAR");
       else if(method == 3)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            outtextxy(x[method]+14,y[method]+12," EDGE");
        else if(method == 4)
            outtextxy(x[method]+14,y[method]+12,"PIXEL");
        else if(method == 5)
            outtextxy(x[method]+14,y[method]+12," SAVE");
        else if(method == 6)
            outtextxy(x[method]+14,y[method]+12," EXIT");

    if(enter == NOT_ENTER)
    {
        setfillstyle(1,24);
        bar(x[method],y[method]+31,x[method]+72,y[method]+32);
        bar(x[method]+71,y[method]+31,x[method]+72,y[method]);
        setfillstyle(1,15);
        bar(x[method]-2,y[method]-2,x[method]-1,y[method]+32);
        bar(x[method]-2,y[method]-2,x[method]+72,y[method]-1);
    }
    else if(enter == ENTER)
    {
        setfillstyle(1,15);
        bar(x[method],y[method]+31,x[method]+72,y[method]+32);
        bar(x[method]+71,y[method]+31,x[method]+72,y[method]);
        setfillstyle(1,24);
        bar(x[method]-2,y[method]-2,x[method]-1,y[method]+32);
        bar(x[method]-2,y[method]-2,x[method]+72,y[method]-1);
    }
}

void oursor(int row,int col)
{
  union REGS ireg;
  ireg.h.ah = 0x02;
  ireg.h.bh = getvpage();
  ireg.h.dh = row - 1;
  ireg.h.dl = col - 1;
  int86(0x10,&ireg,&ireg);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

int getvpage(void)
{
   union REGS ireg;
   ireg.h.ah = 0x0f;
   int86(0x10,&ireg,&ireg);
   return(ireg.h.bh);
}

void filenotfound(void)
{
    char *ch;
    int x = 620 ,y = 400;
    setfillstyle(1,80);
    bar(x,y,x+200,y+100);
    setfillstyle(1,24);
    bar(x,y+101,x+202,y+102);
    bar(x+201,y+101,x+202,y);
    setfillstyle(1,82);
    bar(x-2,y-2,x-1,y+101);
    bar(x-2,y-2,x+202,y-1);
    setcolor(14);
    outtextxy(x+50,y+20,"FILE NOT FOUND");
    strcpy(ch," OK");
    bottom(x+50,450,1,ch);
    getch();
    bottom(x+50,450,0,ch);
    delay(300);
    bottom(x+50,450,1,ch);
    delay(300);
    bottom(x+50,450,0,ch);
    setfillstyle(1,1);
    bar(x-2,y-2,x+202,y+102);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void clearwin(int x1,int y1,int x2,int y2)
{
    setfillstyle(1,0);
    bar(x1,y1,x2,y2);  // fill background for barselect
}

void crosssign(int x,int y)
{
   setcolor(12);
   line(x-5,y,x+5,y);
   line(x,y-5,x,y+5);
}

void clear(int winselect)
{
            if(winselect == 1)
             {
                clearwin(SHOWX,SHOW1Y,SHOWX+MAXC,SHOW1Y+MAXR);
                setfillstyle(1,80);
                bar(620+130+1,550+5+1,620+330-1,550+30-1);
             }
            else if(winselect == 2)
             {
                clearwin(SHOWX,SHOW2Y,SHOWX+MAXC,SHOW2Y+MAXR);
                setfillstyle(1,80);
                bar(620+130+1,620+5+1,620+330-1,620+30-1);
             }
}

int histrogram(char iname[50],int winselect)
{
    int line_y,pel,i,wx = SHOWX,wy;
    long int gray[64];
    FILE *ifp;
    unsigned char *temp = new unsigned char[MAXC];
    ifp=fopen(iname,"r+b");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for(i=0;i<64;i++)
   gray[i] = 0;
if(winselect == 1)
      wy = SHOW1Y;
else
      wy = SHOW2Y;
for(line_y=0;line_y<MAXR;line_y++)
  {
      fread(temp,sizeof(unsigned char),MAXC,ifp);
      for(pel=0;pel<MAXC;pel++)
        {
         if(temp[pel] > 63)
              gray[63]++;
         else
              gray[temp[pel]]++;
        }
  }
delete temp;
fclose(ifp);
setfillstyle(1,11);
bar(wx+1,wy+1,wx+511,wy+255);
setcolor(1);
line(wx+35,wy+240,wx+480,wy+240);
line(wx+35,wy+16,wx+35,wy+240);
long int graymax = gray[0];
for(i=0;i<64;i++)
  {
      if(gray[i] > graymax)
         graymax = gray[i];
  }
long int devide;
int j = 1;
do{
      devide = graymax/j;
      j++;
}while(devide > 224);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
setcolor(3);
for(i=0;i<63;i++)
 {
     line(wx+35+i*7,wy+240-gray[i]/j,
         wx+35+(i+1)*7,wy+240-gray[i+1]/j);
 }
long float t_g=0,g_f=0,m_g;
long float m_g_1,a,b,ab[64];
for(i=0;i<64;i++)
 {
     t_g = gray[i]+t_g;
     g_f = i*gray[i]+g_f;
     ab[i] = 0;
 }
m_g_1 = g_f/t_g;
t_g = 0;
g_f = 0;
for(i=0;i<63;i++)
 {
     t_g = gray[i]+t_g;
     g_f = i*gray[i]+g_f;
     m_g = g_f/t_g;
     b = (m_g-m_g_1)*(m_g-m_g_1);
     a = t_g/(MAXR*MAXC-t_g);
     ab[i] = a*b-1;
 }
long float abmax = ab[0];
int max=0;
for(i=0;i<63;i++)
 {
     if(ab[i] >= abmax)
      {
         abmax = ab[i];
         max = i;
      }
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    line(wx+35+max,wy+16,wx+35+max,wy+240);
    getch();
    return max;
}

/*int thinning(int winselect,int method)
{
    int wx = SHOWX,wy1,wy2,line_x,line_y,i;
    unsigned char value,n,s=0,s2=0,p[8],p2[8];
    if(winselect == 1)
     {
          wy1 = SHOW1Y;
          wy2 = SHOW2Y;
     }
    else
     {
          wy1 = SHOW2Y;
          wy2 = SHOW1Y;
     }
  for(line_y=1;line_y<MAXR-1;line_y++)
    {
          for(line_x=1;line_x<MAXC-1;line_x++)
           {
              value = getpixel(line_x+wx,line_y+wy1)-192;
              switch(value)
               {
                    case 0:
                       {
                            putpixel(line_x+wx,line_y+wy2,192);
                       }
                    break;
                    case 63:

                         switch(method)
                          {
                            case 1:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
p[0] = getpixel(line_x+wx,line_y+wy1-1);
p[1] = getpixel(line_x+wx+1,line_y+wy1-1);
p[2] = getpixel(line_x+wx+1,line_y+wy1);
p[3] = getpixel(line_x+wx+1,line_y+wy1+1);
p[4] = getpixel(line_x+wx,line_y+wy1+1);
p[5] = getpixel(line_x+wx-1,line_y+wy1+1);
p[6] = getpixel(line_x+wx-1,line_y+wy1);
p[7] = getpixel(line_x+wx-1,line_y+wy1-1);
p2[4] = getpixel(line_x+wx,line_y+wy1-1);
p2[5] = getpixel(line_x+wx+1,line_y+wy1-1);
p2[6] = getpixel(line_x+wx+1,line_y+wy1);
p2[7] = getpixel(line_x+wx+1,line_y+wy1+1);
p2[0] = getpixel(line_x+wx,line_y+wy1+1);
p2[1] = getpixel(line_x+wx-1,line_y+wy1+1);
p2[2] = getpixel(line_x+wx-1,line_y+wy1);
p2[3] = getpixel(line_x+wx-1,line_y+wy1-1);
for(i=0;i<8;i++)
 {
   if(p[i] == 192)
         p[i] = 0;
   else
         p[i] = 1;
 }
for(i=0;i<8;i++)
 {
   if(p2[i] == 192)
         p2[i] = 0;
   else
         p2[i] = 1;
 }
n = p[0]+p[1]+p[2]+p[3]+p[4]+p[5]+p[6]+p[7];
s = 0;s2 = 0;
for(i=0;i<7;i++)
 {
   if((p[i] == 0) && (p[i+1] == 1))
         s++;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
      if((p[7] == 0) && (p[0] == 1))
        s++;
      for(i=0;i<7;i++)
        {
          if((p2[i] == 0) && (p2[i+1] == 1))
                s2++;
        }
      if((p2[7] == 0) && (p2[0] == 1))
        s2++;
      if((n >= 2 && n <= 6) && (s == 1) &&
        (((p[0] && p[2] && p[4]) == 0) &&
        ((p[2] && p[4] && p[6]) == 0)))
          putpixel(line_x+wx,line_y+wy2,192);
      else if((n >= 2 && n <=6 ) && (s2 == 1) &&
        (((p2[0] && p[2] && p[4]) == 0) &&
        ((p[2] && p[4] && p[6]) == 0)))
          putpixel(line_x+wx,line_y+wy2,192);
/*    else if(((p[3] + p[4] + p[5]) == 3) && ((p[7]+p[0]+p[1]) == 0))
          putpixel(line_x+wx,line_y+wy2,192);
      else if(((p[7] + p[0] + p[1]) == 3) && ((p[3]+p[4]+p[5]) == 0))
          putpixel(line_x+wx,line_y+wy2,192);
      else if(((p[1] + p[2] + p[3]) == 3) && ((p[5]+p[6]+p[7]) == 0))
          putpixel(line_x+wx,line_y+wy2,192);
      else if(((p[5] + p[6] + p[7]) == 3) && ((p[1]+p[2]+p[3]) == 0))
          putpixel(line_x+wx,line_y+wy2,192):*/
      else
        putpixel(line_x+wx,line_y+wy2,192+63);
    break;
    case 2:
      p[0] = getpixel(line_x+wx,line_y+wy1-1);
      p[1] = getpixel(line_x+wx+1,line_y+wy1-1);
      p[2] = getpixel(line_x+wx+1,line_y+wy1);
      p[3] = getpixel(line_x+wx+1,line_y+wy1+1);
      p[4] = getpixel(line_x+wx,line_y+wy1+1);
      p[5] = getpixel(line_x+wx-1,line_y+wy1+1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
p[6] = getpixel(line_x+wx-1,line_y+wy1);
p[7] = getpixel(line_x+wx-1,line_y+wy1-1);
for(i=0;i<8;i++)
  {
    if(p[i] == 192)
        p[i] = 0;
    else
        p[i] = 1;
  }
if(((p[3]+p[4]+p[5])==3)&&((p[2]+p[6])>=1))
    putpixel(line_x+wx,line_y+wy2,192);
else if(((p[7]+p[0]+p[1])==3)&&((p[6]+p[2])>=1))
    putpixel(line_x+wx,line_y+wy2,192);
else
   putpixel(line_x+wx,line_y+wy2,192+63);
break;
case 3:
p[0] = getpixel(line_x+wx,line_y+wy1-1);
p[1] = getpixel(line_x+wx+1,line_y+wy1-1);
p[2] = getpixel(line_x+wx+1,line_y+wy1);
p[3] = getpixel(line_x+wx+1,line_y+wy1+1);
p[4] = getpixel(line_x+wx,line_y+wy1+1);
p[5] = getpixel(line_x+wx-1,line_y+wy1+1);
p[6] = getpixel(line_x+wx-1,line_y+wy1);
p[7] = getpixel(line_x+wx-1,line_y+wy1-1);
for(i=0;i<8;i++)
  {
    if(p[i] == 192)
        p[i] = 0;
    else
        p[i] = 1;
  }
if(((p[1] + p[2] + p[3]) == 3) && ((p[0]+p[4]) >= 1))
    putpixel(line_x+wx,line_y+wy2,192);
else if(((p[5] + p[6] + p[7]) == 3) && ((p[0]+p[4]) >= 1))
    putpixel(line_x+wx,line_y+wy2,192);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน[illegible]ให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*                                  else if((p[6] == 0) && (p[2] == 1))
                                        putpixel(line_x+wx,line_y+wy2,192);
                                    else if((p[6] == 1) && (p[2] == 0))
                                        putpixel(line_x+wx,line_y+wy2,192);*/
                                    else
                                      putpixel(line_x+wx,line_y+wy2,192+63);
                                break;
                            }
                        break;
                    }
                }
            }
        int result = compare();  // result == 1 :- not same
        if(result == 1)
            {
                clear(winselect);
                switch(winselect)
                    {
                        case 1:  return 2; break;
                        case 2:  return 1; break;
                    }
            }
        else
            {
                clear(winselect);
                switch(winselect)
                    {
                      case 1: return 4; break;
                      case 2: return 3; break;
                    }
            }
    } */

    int compare(void)
    {
      int line_x,line_y,w1,w2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  for(line_x=1;line_x<511;line_x++)
   {
       for(line_y=1;line_y<255;line_y++)
         {
            w1 = getpixel(line_x+SHOWX,line_y+SHOW1Y);
            w2 = getpixel(line_x+SHOWX,line_y+SHOW2Y);
            if(w2 != w1)
                return 1; // two pioture not same
         }
   }
  return 0;
}

void kmit_song(void)
{
#define  C  262
#define  D  294
#define  E  330
#define  F  350
#define  G  392
#define  A  440
#define  B  467
#define  CH 524
#define  DH 588
#define  EH 660
#define  FH 700
#define  GH 785
#define  AH 880
#define  BH 934
#define  REST 25000
#define  speed 1.556
#define  QQE 937.5
#define  Q 375
#define  EN 187.5
#define  ES 287.25
#define  S 93.75
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define EP 187.5
#define QP 375
#define BLANK 5
#define MAXNOTE 109
        unsigned int key=0x0000;
        int i=0;
        char dummy;
        int melody[] = {A,DH,CH,A,G,F,D,G,F,D,C,A,DH,CH,G,A,DH,CH,FH,DH,A,CH,F,G,A,G,
                                C,A,G,F,D,C,F,G,A,G,A,CH,G,A,G,A,DH,CH,A,REST,A,G,D,F,

        B,CH,B,CH,REST,CH,REST,DH,CH,B,CH,A,G,F,REST,G,REST,G,REST,G,C,G,CH,A,G,E,D,E,G
,E,D,C,REST,

        A,DH,CH,A,G,F,D,G,F,D,C,A,DH,CH,G,A,DH,CH,FH,DH,A,CH,A,G,D,F};
        int del[] =
{EN,EN,QQE,EN,EN,EN,EN,EN,EN,EN,Q,EN,EN,ES,S,EN,EN,EN,EN,EN,EN,EN,EN,EN,EN,Q,

        EN,EN,QQE,EN,EN,EN,ES,S,ES,S,Q,EN,EN,ES,S,EN,EN,EN,S,BLANK,S,EN,EN,QQE,

        ES,S,EN,Q,20,Q,EP,ES,S,EN,EN,EN,EN,EN,EP,S,BLANK,S,BLANK,EN,EN,EN,EN,EN,EN,ES,S,
,EN,EN,EN,EN,Q,QP,

        EN,EN,QQE,EN,EN,EN,EN,EN,EN,EN,Q,EN,EN,ES,S,EN,EN,EN,EN,EN,EN,EN,EN,EN,EN,QQE
};
        do
        {
                sound(melody[i]);
                delay(del[i]*speed);
                if(i<MAXNOTE) i++;
                if(i==MAXNOTE)
                {
                        nosound();
                        sleep(2);
                        i=0;
                }
                key=readkey();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }while(key == 0xffff);
        nosound();
        create_border(SHOWX+5,SHOW1Y+5,SHOWX+MAXC-5,SHOW1Y+MAXR-5,0);
}

int readkey(void)
{
        unsigned int sorsave_o1=0,sorsave_o2=0;
        _AH=0x11;
        geninterrupt(0x16);
        if(!(_FLAGS&64))
        {
                sorsave_o1=sorsave_o2=0;
                _AH=0x10;
                geninterrupt(0x16);
                return(_AX);
        }
        sorsave_o1++;
        if(sorsave_o1==5500) {sorsave_o1=0;sorsave_o2++;}
        return 0xffff;
}

void showtitle(void)
{
  int *show;
  create_border(SHOWX+5,SHOW1Y+5,SHOWX+MAXC-5,SHOW1Y+MAXR-5,13);
  create_border(SHOWX+30,SHOW1Y+30,SHOWX+MAXC-30,SHOW1Y+MAXR-30,2);
  settextstyle(GOTHIC_FONT, HORIZ_DIR,3);
  outtextxy(SHOWX+100,SHOW1Y+30,"Stereo  Vision  V.1.00a");
  outtextxy(SHOWX+100,SHOW1Y+65,"       Produced by");
  outtextxy(SHOWX+90,SHOW1Y+90,"Control Engineering Department");
  outtextxy(SHOWX+40,SHOW1Y+115,"King Mongkut 's Institute of Theohonology");
  outtextxy(SHOWX+40,SHOW1Y+140,"         Ladkrabang (KMITL)");
  settextstyle(DEFAULT_FONT,HORIZ_DIR,1);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void create_border(int fx,int fy,int lx,int ly,int color)
{
  int  afx, afy, alx, aly;
  setfillstyle(1,color);
    afx=(lx-fx)/2+fx;
    afy=(ly-fy)/2+fy;
    alx=afx; aly=afy;
  do{
     bar(afx,afy,alx,aly);
     if (afx>fx) { afx-=5; }
     if (afy>fy) { afy-=5; }
     if (alx<lx) { alx+=5; }
     if (aly<ly) { aly+=5; }
  }
  while((afx>fx)||(afy>fy)||(alx<lx)||(aly<ly));
}

void selectpixel(int winselect)
{
 int x,y,i,j,ys,xs,togle;
 unsigned gx[11],gy[11],key=0x0000;
 x = SHOWX + MAXC/2;
 if(winselect == 1)
  {
    y = SHOW1Y+MAXR/2;
    ys = SHOW1Y;
  }
 else if(winselect == 2)
  {
    y = SHOW2Y+MAXR/2;
    ys = SHOW2Y;
  }
 for(i=0;i<11;i++)
   gx[i] = getpixel(x-5+i,y);
 for(i=0;i<11;i++)
   gy[i] = getpixel(x,y-5+i);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
crosssign(x,y);
displayvalue(x,y,winselcot);
do{
 key=bioskey(0);
 for(i=0;i<11;i++)
   putpixel(x-5+i,y,gx[i]);
 for(i=0;i<11;i++)
   putpixel(x,y-5+i,gy[i]);
 if(key == SPACEBAR)
  {
       if(togle == 0)
         togle = 1;
       else // togle = 1
         togle = 0;
  }
 else if(key == Left_Key)
  {
       if(togle == 0)
         x = x-1;
       else // togle == 1
        {
          if(x > SHOWX+50)
                x = x-50;
          else if((x > SHOWX) && (x < SHOWX+50))
                x = SHOWX;
        }
       displayvalue(x,y,winselcot);
  }
 else if(key == Right_Key)
  {
       if(togle == 0)
         x = x+1;
       else
        {
              if(x < SHOWX+MAXC-50)
              x = x+50;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                else if(x > SHOWX+MAXC-50)
                  x = SHOWX+MAXC-1;
            }
          displayvalue(x,y,winselect);
    }
  else if(key == Up_Key)
    {
          if(toglc == 0)
            y = y-1;
          else
            {
              if(y > ys+50)
                    y = y-50;
              else if((y > ys) && (y < ys+50))
                    y = ys;
            }
          displayvalue(x,y,winselect);
    }
  else if(key == Down_Key)
    {
          if(toglc == 0)
            y = y+1;
          else
            {
                  if(y < ys+MAXR-50)
                    y = y+50;
                  else if(y > ys+MAXC-50)
                    y = ys+MAXR-1;
            }
          displayvalue(x,y,winselect);
    }
  for(i=0;i<11;i++)
      gx[i] = getpixel(x-5+i,y);
  for(i=0;i<11;i++)
      gy[i] = getpixel(x,y-5+i);
  orosssign(x,y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }while((key != Eso_Key) && (key != Enter_Key));
  displayvalue(x,y,winselcot);
  switch(key)
   {
       case Eso_Key:for(i=0;i<11;i++)
                      putpixel(x-5+i,y,gx[i]);
                    for(i=0;i<11;i++)
                      putpixel(x,y-5+i,gy[i]);
                    setfillstyle(1,80);
                    switch(winselcot)
                      {
                          case 1:bar(752,557,948,578);
                                 break;
                          case 2:bar(748,623,948,648);
                                 break;
                      }
            break;
       case Enter_Key:
            break;
   }
}


void displayvalue(int x,int y,int winselcot)
{
  char st1[25],st2[25];
  setfillstyle(1,80);
  x = x-SHOWX-MAXC/2;
  itoa(x,st1,10);
  if(winselcot == 1)
   {
     bar(752,557,948,578);
     outtextxy(770,563,"X = ");
     outtextxy(850,563,"Y = ");
     y = -(y - SHOW1Y - MAXR/2);
     itoa(y,st2,10);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        outtextxy(800,563,st1);
        outtextxy(880,563,st2);
      }
    else if(winselect == 2)
      {
        bar(748,623,948,648);
        outtextxy(770,633,"X = ");
        outtextxy(850,633,"Y = ");
        y = -(y - SHOW2Y-MAXR/2);
        itoa(y,st2,10);
        outtextxy(800,633,st1);
        outtextxy(880,633,st2);
      }
}

void savepicture(char iname[50],int winselect)
{
    int wx = SHOWX,wy,line_x,line_y;
    FILE *fp;
    unsigned char value;
    if(winselect == 1)
          wy = SHOW1Y;
    else
          wy = SHOW2Y;
    if((fp=fopen(iname,"w+b"))==NULL)
      exit(1);
    else
      {
            for(line_y=0;line_y<256;line_y++)
              {
                for(line_x=0;line_x<512;line_x++)
                    {
                      value = getpixel(wx+line_x,wy+line_y)-192;
                      fwrite(&value,sizeof(unsigned char),1,fp);
                    }
              }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
  fclose(fp);
}

void zeroorossing(int sigma,int k,int rando,
                  int x1,int y1,int x2,int y2,int winselect)
{
  int w1,w2,i,j,h1[11],h2[11],g[11][11],line_y,line_x;
  long int *line1 = new long int[59];
  long int *line2 = new long int[59];
  long int *line3 = new long int[59];
  long int *line4 = new long int[59];
  long int *line5 = new long int[59];
  long int a[13];
  int x,y;
  long int color[121],col;
  int locatex,locatey;
  FILE *fp,*ifp;
  FILE *datafile,*coordinate;
  struct datatype{
                  long int mag;
                  int direct;
                  int sign;
                 }data;
  char iname1[50];
  char iname2[50];
  if((fp=fopen("data.tmp","wb"))== NULL)
    exit(1);
  if(winselect == 1)
   {
     w1 = SHOW1Y;
     w2 = SHOW2Y;
   }
  else if(winselect == 2)
   {
     w1 = SHOW2Y;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    w2 = SHOW1Y;
  }
for(i=0;i<(rando-1)/2+1;i++)
 {
   h1[(rando-1)/2-i] = sqrt(k)*(1-(i*i)/(sigma*sigma))*exp(-(i*i)/(2*(sigma*sigma)));
   h2[(rando-1)/2-i] = sqrt(k)*exp(-(i*i)/(2*(sigma*sigma)));
 }
for(i=0;i<(rando-1)/2;i++)
 {
   h1[i+(rando-1)/2+1] = h1[(rando-1)/2-i-1];
   h2[i+(rando-1)/2+1] = h2[(rando-1)/2-i-1];
 }
for(i=0;i<rando;i++)
 {
  for(j=0;j<rando;j++)
        g[i][j] = h1[j]*h2[i] + h1[i]*h2[j];
 }
for(linc_y=(rando-1)/2+y1+1;linc_y<y2-(rando-1)/2;linc_y++)
 {
      for(linc_x=(rando-1)/2+x1+1;linc_x<x2-(rando-1)/2;linc_x++)
        {
          for(i=0;i<rando;i++)
            {
              for(j=0;j<rando;j++)
                {
                  locatex = (SHOWX+linc_x)-(rando-1)/2;
                  locatey = (w1+linc_y)-(rando-1)/2;
                  color[i*rando+j] = (getpixel(locatex+i,locatey+j)-192)
                                                *g[i][j];
                }
            }
          col = color[0];
          for(i=1;i<rando*rando;i++)
              col = col+color[i];
          if(col > 0)
                putpixel(SHOWX+linc_x,w2+linc_y,9);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            else if(ool == 0)
                putpixel(SHOWX+line_x,w2+line_y,2);
            else if(ool < 0)
                putpixel(SHOWX+line_x,w2+line_y,12);
            fwrite(&ool,sizeof(long int),1,fp);
        }
    }
    fclose(fp);
    clearwin(SHOWX+x1+1,w1+y1+1,SHOWX+x2-1,w1+y2-1);
    if((ifp=fopen("data.tmp","rb"))== NULL)
        exit(1);
    switch(winselect)
    {
        case 1:strcpy(iname1,"data1.dat");
               strcpy(iname2,"cor1.cor");
               break;
        case 2:strcpy(iname1,"data2.dat");
               strcpy(iname2,"cor2.cor");
               break;
    }
    if((cordinate = fopen(iname2,"wb"))==NULL)
      exit(1);
    fwrite(&x1,sizeof(int),1,cordinate);
    fwrite(&y1,sizeof(int),1,cordinate);
    fclose(cordinate);
    if((datafile = fopen(iname1,"wb"))==NULL)
      exit(1);
    for(line_y=(rando-1)/2+y1+1;line_y<y2-(rando-1)/2-2;line_y++)
      {
            if(line_y == (rando-1)/2+y1+1)
                fread(line1,sizeof(long int),59,ifp);
            if(line_y == (rando-1)/2+y1+1+1)
                fread(line2,sizeof(long int),59,ifp);
            if(line_y == (rando-1)/2+y1+1+2)
                fread(line3,sizeof(long int),59,ifp);
            if(line_y == (rando-1)/2+y1+1+3)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        fread(line4,sizeof(long int),59,ifp);
    if(line_y == (rando-1)/2+y1+1+4)
      {
        fread(line5,sizeof(long int),59,ifp);
        for(line_x=(rando-1)/2+x1+1;line_x<x2-(rando-1)/2;
              line_x++)
            {
              if((line_x>=(rando-1)/2+x1+3)&&
                 (line_x<x2-(rando-1)/2-2))
                    {
                      a[0]=line2[line_x-(rando-1)/2-x1-1-1];//a
                      a[1]=line2[line_x-(rando-1)/2-x1-1];  //b
                      a[2]=line2[line_x-(rando-1)/2-x1-1+1];//c
                      a[3]=line3[line_x-(rando-1)/2-x1-1-1];//d
                      a[4]=line3[line_x-(rando-1)/2-x1-1];  //e
                      a[5]=line3[line_x-(rando-1)/2-x1-1+1];//f
                      a[6]=line4[line_x-(rando-1)/2-x1-1-1];//g
                      a[7]=line4[line_x-(rando-1)/2-x1-1];  //h
                      a[8]=line4[line_x-(rando-1)/2-x1-1+1];//i
                      a[9]=line1[line_x-(rando-1)/2-x1-1];  //l
                      a[10]=line3[line_x-(rando-1)/2-x1-1-2];//q
                      a[11]=line3[line_x-(rando-1)/2-x1-1+2];//r
                      a[12]=line5[line_x-(rando-1)/2-x1-1];  //w
                        data.mag = a[4];
                        data.direct=comparedirect(a,winselect,
                                        line_x,line_y);
                      if(a[4] > 0)
                            data.sign = 1;
                      else if(a[4] == 0)
                            data.sign = 0;
                      else if(a[4] < 0)
                            data.sign = -1;
                      fwrite(&data,sizeof(struct datatype),1,
                            datafile);
                    }
            }
      }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
  else
    {
      line1 = line2;
      line2 = line3;
      line3 = line4;
      line4 = line5;
      fread(line5,sizeof(long int),59,ifp);
      for(linc_x=(rando-1)/2+x1+1;linc_x<x2-(rando-1)/2;
          linc_x++)
        {
          if((linc_x>(rando-1)/2+x1+3)&&
             (linc_x<x2-(rando-1)/2-3))
               {
                 a[0]=linc2[linc_x-(rando-1)/2-x1-1-1];//a
                 a[1]=linc2[linc_x-(rando-1)/2-x1-1];  //b
                 a[2]=linc2[linc_x-(rando-1)/2-x1-1+1];//c
                 a[3]=linc3[linc_x-(rando-1)/2-x1-1-1];//d
                 a[4]=linc3[linc_x-(rando-1)/2-x1-1];  //e
                 a[5]=linc3[linc_x-(rando-1)/2-x1-1+1];//f
                 a[6]=linc4[linc_x-(rando-1)/2-x1-1-1];//g
                 a[7]=linc4[linc_x-(rando-1)/2-x1-1];  //h
                 a[8]=linc4[linc_x-(rando-1)/2-x1-1+1];//i
                 a[9]=linc1[linc_x-(rando-1)/2-x1-1];  //l
                 a[10]=linc3[linc_x-(rando-1)/2-x1-1-2];//q
                 a[11]=linc3[linc_x-(rando-1)/2-x1-1+2];//r
                 a[12]=linc5[linc_x-(rando-1)/2-x1-1];  //w
                   data.mag = a[4];
                   data.direct=comparedirect(a,winselect,
                                linc_x,linc_y);
                   if(a[4] > 0)
                         data.sign = 1;
                   else if(a[4] == 0)
                         data.sign = 0;
                   else if(a[4] < 0)
                         data.sign = -1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                    fwrite(&data,sizeof(struct datatype),1,
                                                datafile);
                                }
                            }
                        }
                    }
                fclose(ifp);
                fclose(datafile);
                delete line1;
                delete line2;
                delete line3;
                delete line4;
                delete line5;
}

int comparedirect(long int z[],int winselect,int x,int y)
{
  int direct,locatex,locatey,w1;
  if(winselect == 1)
     w1 = SHOW1Y;
  else if(winselect == 2)
     w1 = SHOW2Y;
  locatex = SHOWX+x;
  locatey = w1+y;
  if(z[4] == 0)  // Predicate A - D
    {
       if((z[3] > 0) && (z[5] < 0))// Predicate A
         {
           if((z[1] > 0) && (z[7] < 0))
               {
                 direct = 6;
                 putpixel(locatex,locatey,6);
               }
           else if((z[1] < 0) && (z[7] > 0))
               {
                 direct = 8;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            putpixel(locatex,locatey,8);
           }
      else if(((z[1] > 0) && (z[7] > 0)) || ((z[1] < 0)&&(z[7]<0)))
           {
            direct = 7;
            putpixel(locatex,locatey,7);
           }
   }
 else if((z[3] < 0) && (z[5] > 0))  // Predicate B
   {
     if((z[1] > 0) && (z[7] < 0))
          {
           direct = 4;
           putpixel(locatex,locatey,4);
          }
     else if((z[1] < 0) && (z[7] > 0))
          {
           direct = 2;
           putpixel(locatex,locatey,2);
          }
     else if(((z[1] < 0) && (z[7]<0))||((z[1] > 0) && (z[7] > 0)))
          {
           direct = 7;
           putpixel(locatex,locatey,7);
          }
   }
 else if((z[1] > 0) && (z[7] < 0))  // Predicate C
   {
     if(((z[3] > 0) && (z[5] > 0)) || ((z[3]<0)&&(z[5] < 0)))
          {
           direct = 5;
           putpixel(locatex,locatey,5);
          }
   }
 else if((z[1] < 0) && (z[7] > 0))  // Predicate D
   {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            if(((z[5]< 0) && (z[3] < 0)) ||((z[5] > 0) && (z[3] > 0)))
                {
                 direct = 1;
                 putpixel(locatex,locatey,1);
                }
          }
    }
else // Predicate E - K
    {
        if((abs(z[4])<z[1])&&(z[9]>0)&&(z[0]<0)&&
          (z[2]>0)&&(z[4]<0)&&(z[1]>0))//Predicate E.1
          {
            direct = 4;
            putpixel(locatex,locatey,4);
          }
        else if((abs(z[4])<z[5])&&(z[4]<0)&&(z[2]>0)&&
                (z[11]>0)&&(z[8]<0)&&(z[5]>0)) //Predicate E.2
        {
          direct = 4;
          putpixel(locatex,locatey,4);
        }
        else if((z[1]>0)&&(z[3]<0)&&(z[5]>0)&&
                (z[7]<0)&&(z[4]>0)) // Predicate E.3
          {
            direct = 4;
            putpixel(locatex,locatey,4);
          }
        else if((abs(z[4])<z[7])&&(z[6]<0)&&(z[12]>0)&&
                (z[9]>0)&&(z[4]<0)&&(z[7]>0)) // Predicate E.4
          {
            direct = 2;
            putpixel(locatex,locatey,2);
          }
        else if((abs(z[4])<z[5])&&(z[4]<0)&&(z[2]<0)&&
                (z[11]>0)&&(z[8]>0)&&(z[5]>0)) // Predicate E.5
          {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    direct = 2;
    putpixel(locatex,locatey,2);
  }
else if((z[4]>0)&&(z[1]<0)&&(z[3]<0)&&
        (z[7]>0)&&(z[5]>0)) // Predicate E.6
  {
    direct = 2;
    putpixel(locatex,locatey,2);
  }
else if((abs(z[4])<z[5])&&(z[4]<0)&&(z[5]>0)&&
        (z[2]>0)&&(z[11]>0)&&(z[8]>0))//Pre E.7
  {
    direct = 3;
    putpixel(locatex,locatey,3);
  }
else if((z[3]<0)&&(z[5]>0))          // Pre E.8
  {
    direct = 3;
    putpixel(locatex,locatey,3);
  }
else if((abs(z[4])<z[1])&&(z[1]>0)&&(z[4]<0)&&
        (z[0]<0)&&(z[2]<0)&&(z[9]>0)) // Pre F.1
  {
    direct = 4;
    putpixel(locatex,locatey,4);
  }
else if((z[4]>0)&&(z[3]<0)&&(z[5]<0)&&(z[1]>0)) // Pre  F.2
  {
    direct = 5;
    putpixel(locatex,locatey,5);
  }
else if((z[4]<0)&&(z[3]>0)&&(z[10]<0)&&(z[0]>0)) // Pre F.3
  {
    direct = 5;
    putpixel(locatex,locatey,5);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else if((z[4]>0)&&(z[3]<0)&&(z[1]<0)&&
        (z[5]<0)&&(z[7]>=0)) // Pre F.4
  {
    direct = 1;
    putpixel(looatex,looatey,1);
  }
else if((z[4]<0)&&(z[3]>0)&&(z[10]<0)&&
        (z[0]<0)&&(z[6]>=0)) // Pre F.5
  {
    direct = 1;
    putpixel(looatex,looatey,1);
  }
else if((abs(z[4])<z[5])&&(z[4]<0)&&
        (z[5]>0)&&(z[1]<0)) // Pre F.6
  {
    direct = 3;
    putpixel(looatex,looatey,3);
  }
else if((z[4]>0)&&(z[3]<0)&&(z[5]<0)) // Pre F.7
  {
    direct = 3;
    putpixel(looatex,looatey,3);
  }
else if((z[4]<0)&&(z[3]>0)&&(z[10]<0)) // Pre F.8
  {
    direct = 7;
    putpixel(looatex,looatey,7);
  }
else if((abs(z[4])<z[7])&&(z[4]<0)&&(z[7]>0)&&
        (z[6]>0)&&(z[12]>0)&&(z[8]>0)) // Pre G.1
  {
    direct = 1;
    putpixel(looatex,looatey,1);
  }
else if((abs(z[4])<z[7])&&(z[4]<0)&&(z[7]>0)&&
        (z[6]>0)&&(z[12]>0)&&(z[8]<0)) // Pre G.2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
    direct = 8;
    putpixel(looatex,looatey,8);
  }
else if((z[3]>0)&&(z[10]>0)&&(z[6]>0)&&(z[0]<0)&&
        (z[4]<0)&&(abs(z[4])<z[3])) // Pre G.3
  {
    direct = 8;
    putpixel(looatex,looatey,8);
  }
else if((z[1]>0)&&(z[10]>0)&&(z[6]>0)&&(z[0]<0)&&
        (z[4]<0)&&(abs(z[4])>z[3])) // Pre G.4
  {
    direct = 8;
    putpixel(looatex,looatey,8);
  }
else if((z[4]>0)&&(z[7]>0)&&(z[1]<0)) // Pre G.5
  {
    direct = 1;
    putpixel(looatex,looatey,1);
  }
else if((z[9]<0)&&(z[1]>0)&&(z[2]<0)&&(z[4]<0)&&(z[3]==0))//Pre H.1
  {
    direct = 6;
    putpixel(looatex,looatey,6);
  }
else if((z[9]<0)&&(z[1]>0)&&(z[2]<0)&&(z[4]<0)&&
        (z[3]>0)) // Pre H.2
  {
    direct = 7;
    putpixel(looatex,looatey,7);
  }
else if((z[1]<0)&&(z[4]>0)&&(z[7]<0)) // Pre H.3
  {
    direct = 7;
    putpixel(looatex,looatey,7);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
 else if((z[1]>0)&&(z[9]<0)) // Pre H.4
  {
    direct = 5;
    putpixel(locatex,locatey,5);
  }
 else if((z[4]<0)&&(z[3]>0)&&(z[6]>0)&&
          (abs(z[4])<z[3])) // Pre I.1
  {
    direct = 7;
    putpixel(locatex,locatey,7);
  }
 else if((z[4]>0)&&(z[7]>0)&&(z[5]<0)) // Pre I.2
  {
    direct = 7;
    putpixel(locatex,locatey,7);
  }
 else if((z[4]<0)&&(z[1]>0)&&(z[2]<0)&&(abs(z[4])<z[1])) // Pre J.1
  {
    direct = 5;
    putpixel(locatex,locatey,5);
  }
 else if((z[4]<0)&&(z[3]>0)&&(z[6]<0)&&(z[1]>0)&&
          (abs(z[4])<z[3])) // Pre J.2
  {
    direct = 6;
    putpixel(locatex,locatey,6);
  }
 else if((z[4]<0)&&(z[3]>0)&&(z[6]<0)&&(z[1]<0)&&
          (abs(z[4])<z[3])) // Pre J.3
  {
    direct = 7;
    putpixel(locatex,locatey,7);
  }
 else if((z[4]<0)&&(z[1]>0)&&(z[2]<0)&&(z[3]>0)) // Pre J.4
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                direct = 6;
                putpixel(looatex,looatey,6);
              }
            else if((z[4]>0)&&(z[5]<0)&&(z[7]<0)) // Pre J.5
              {
                direct = 6;
                putpixel(looatex,looatey,6);
              }
            else if((z[4]>0)&&(z[5]<0)&&(z[7]<0)&&(z[2]>0)) // Pre J.6
              {
                direct = 6;
                putpixel(looatex,looatey,6);
              }
            else if((z[4]>0)&&(z[5]<0)&&(z[7]<0)&&(z[3]<0)) // Pre J.7
              {
                direct = 7;
                putpixel(looatex,looatey,7);
              }
            else if((z[4]<0)&&(z[9]>0)&&(z[0]>0)&&(z[1]>0)&&
                    (z[2]>0)&&(abs(z[4])<z[1])) // Pre K.1
              {
                direct = 5;
                putpixel(looatex,looatey,5);
              }
            else if((z[1]>0)&&(z[3]>0)&&(z[5]>0)&&(z[7]<0)) // Pre K.2
              {
                direct = 5;
                putpixel(looatex,looatey,5);
              }
            else
              {
                direct = 0;
                putpixel(looatex,looatey,13);
              }
        }
    return(direct);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

void selectboundary(int winselect)
{
  int x,y,i,j,ys,x1,y1,x2,y2,togle=0;
  unsigned gx[11],gy[11],key = 0x0000;
  x = SHOWX + MAXC/2;
  if(winselect == 1)
   {
     y = SHOW1Y+MAXR/2;
     ys = SHOW1Y;
   }
  else if(winselect == 2)
   {
     y = SHOW2Y+MAXR/2;
     ys = SHOW2Y;
   }
  for(i=0;i<11;i++)
    gx[i] = getpixel(x-5+i,y);
  for(i=0;i<11;i++)
    gy[i] = getpixel(x,y-5+i);
  crosssign(x,y);
  displayvalue(x,y,winselect);
  do{
    key=bioskey(0);
    for(i=0;i<11;i++)
      putpixel(x-5+i,y,gx[i]);
    for(i=0;i<11;i++)
      putpixel(x,y-5+i,gy[i]);
    if(key == SPACEBAR)
      {
          if(togle == 0)
            togle = 1;
          else // togle = 1
            togle = 0;
      }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else if(key == Left_Key)
 {
      if(togle == 0)
        x = x-1;
      else // togle == 1
        {
          if(x > SHOWX+50)
                x = x-50;
          else if((x > SHOWX) && (x < SHOWX+50))
                x = SHOWX;
        }
      displayvalue(x,y,winselect);
 }
else if(key == Right_Key)
 {
      if(togle == 0)
        x = x+1;
      else
        {
          if(x < SHOWX+MAXC-50)
            x = x+50;
          else if(x > SHOWX+MAXC-50)
            x = SHOWX+MAXC-1;
        }
      displayvalue(x,y,winselect);
 }
else if(key == Up_Key)
 {
      if(togle == 0)
        y = y-1;
      else
        {
          if(y > ys+50)
                y = y-50;
          else if((y > ys) && (y < ys+50))
                y = ys;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            }
            displayvalue(x,y,winselcot);
        }
        else if(key == Down_Key)
        {
            if(togle == 0)
              y = y+1;
            else
            {
                  if(y < ys+MAXR-50)
                    y = y+50;
                  else if(y > ys+MAXC-50)
                    y = ys+MAXR-1;
            }
            displayvalue(x,y,winselcot);
        }
        for(i=0;i<11;i++)
            gx[i] = getpixel(x-5+i,y);
        for(i=0;i<11;i++)
            gy[i] = getpixel(x,y-5+i);
        orosssign(x,y);
        if(key == Enter_Key)
        {
            x1 = x;
            y1 = y;
            x2 = x+70;
            y2 = y+70;
            key = Eso_Key;
            displayvalue(x,y,winselcot);
        }
    }while(key != Eso_Key);
    for(i=0;i<11;i++)
      putpixel(x-5+i,y,gx[i]);
    for(i=0;i<11;i++)
      putpixel(x,y-5+i,gy[i]);
    displayvalue(x,y,winselcot);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    setcolor(13);
    line(x1,y1,x2,y1);
    line(x1,y1,x1,y2);
    line(x2,y1,x2,y2);
    line(x1,y2,x2,y2);
    zerocrossing(1,170,11,x1-SHOWX,y1-ys,x2-SHOWX,y2-ys,winselect);
}

void matching(void)
{
    char iname1[50],iname2[50]; // for the first picture
    char iname3[50],iname4[50]; // for the second picture
    FILE *data1,*cordinate1;
    FILE *data2,*cordinate2;
    int x,y,x1,y1,x2,y2;
    int line_x,line_y;
    int rando = 1;
    int i,j,k,l;
    long int delx1=0,delx2=0;
    struct datatype{
                        long int mag;
                        int direct;
                        int sign;
                    }d1[55][55],d2[55][55];
    strcpy(iname1,"data1.dat");
    strcpy(iname2,"cor1.cor");
    strcpy(iname3,"data2.dat");
    strcpy(iname4,"cor2.cor");
    if((cordinate1=fopen(iname2,"rb"))== NULL)
      exit(1);
    fread(&x1,sizeof(int),1,cordinate1);
    fread(&y1,sizeof(int),1,cordinate1);
    fclose(cordinate1);
    if((cordinate2=fopen(iname4,"rb"))== NULL)
          exit(1);
    fread(&x2,sizeof(int),1,cordinate2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
fread(&y2,sizeof(int),1,oordinate2);
folose(oordinate2);
if((data1=fopen(iname1,"rb"))== NULL)
        exit(1);
i = 0;
for(line_y=(rando-1)/2+y1+3;line_y<(y1+70)-(rando-1)/2-2;line_y++)
  {
       j = 0;
       for(line_x=(rando-1)/2+x1+3;line_x<(x1+70)-(rando-1)/2-2;
           line_x++)
         {
           fread(&d1[i][j],sizeof(struct datatype),1,data1);
           j++;
         }
       i++;
      }
folose(data1);
if((data2=fopen(iname3,"rb"))== NULL)
       exit(1);
i = 0;
for(line_y=(rando-1)/2+y2+3;line_y<(y2+70)-(rando-1)/2-2;line_y++)
  {
       j = 0;
       for(line_x=(rando-1)/2+x2+3;line_x<(x2+70)-(rando-1)/2-2;
           line_x++)
           {
            fread(&d2[i][j],sizeof(struct datatype),1,data2);
            j++;
           }
       i++;
  }
folose(data2);
for(i=0;i<55;i++)
  {
       for(j=0;i<55;i++)
          {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                for(k=0;k<55;k++)
                    {
                      for(l=0;l<55;l++)
                       {
                            if((d1[i][j].direct == d2[k][l].direct) &&
                              (d1[i][j].sign == d2[k][l].sign))
                                  {
                                    if((k == 0) && (l == 0))
                                       delx1 = abs(d1[i][j].mag-d2[k][l].mag);
                                    else
                                      {
                                          if(abs(d1[i][j].mag-d2[k][l].mag)
                                            <delx1)
                                            delx1 = abs(d1[i][j].mag-
                                                    d2[k][l].mag);
                                      }
                                  }
                       }
                      if((i==0) && (j == 0))
                        delx2 = delx1;
                      else
                      {
                            if(delx1<delx2)
                              delx2 = delx1;
                      }
                    }
            }
    }
  x = j+SHOWX+(rando-1)/2+3+x1;
  y = i+SHOW1Y+(rando-1)/2+3+y1;
  crosssign(x,y);
  x = l+SHOWX+(rando-1)/2+3+x2;
  y = i+SHOW2Y+(rando-1)/2+3+y2;
  crosssign(x,y);
  i = i+(rando-1)/2+3-MAXR/2+y1;
  j = j+(rando-1)/2+3-MAXC/2+x1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    k = k+(rando-1)/2+3-MAXR/2+y2;
    l = l+(rando-1)/2+3-MAXC/2+x2;
    displayvalue(j,i,1);
    displayvalue(l,k,2);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

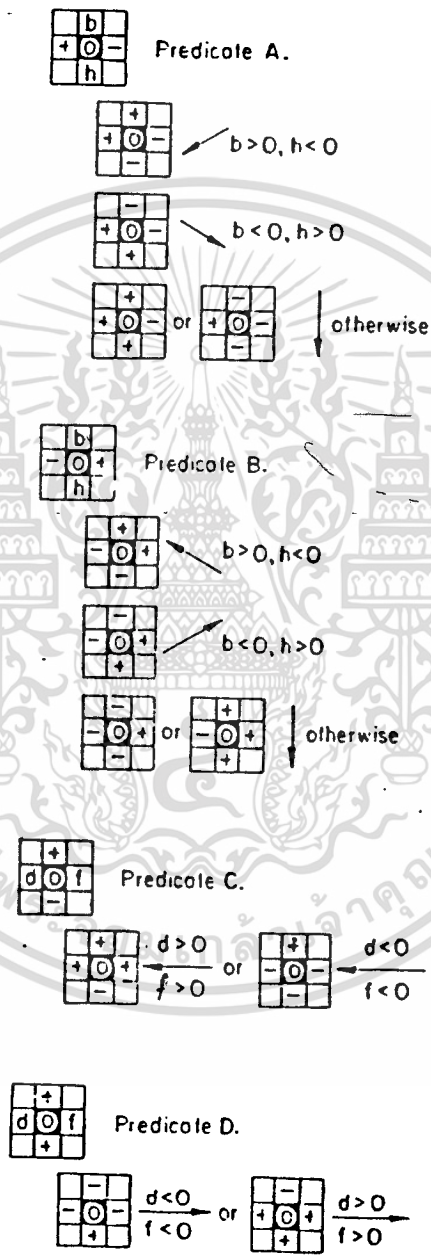
เมื่อใช้อัลกอริทึม Zero- crossing แบบ 2 มิติขนาด 3 x 3 ทำการ convolution กับจุดของภาพ ผลจากการคำนวณจะมีทั้งหมด 11 แบบใหญ่ ๆ (กลุ่ม A - K) โดยเครื่องหมายบวกและลบในแต่ละเซลจะแทนค่าบวกและลบในเซลนั้น เมื่อกำหนดให้ค่าในแต่ละเซลของเมตริกซ์ขนาด 3 x 3 เป็นดังนี้

a	b	c
d	e	f
g	h	i

ค่าเซลที่ถูกล้อมรอบด้วยวงกลมจะหมายถึงตำแหน่งที่ถูกทดสอบและมีทิศทางในแต่ละแบบแสดงไว้ดังลูกศรข้าง ๆ

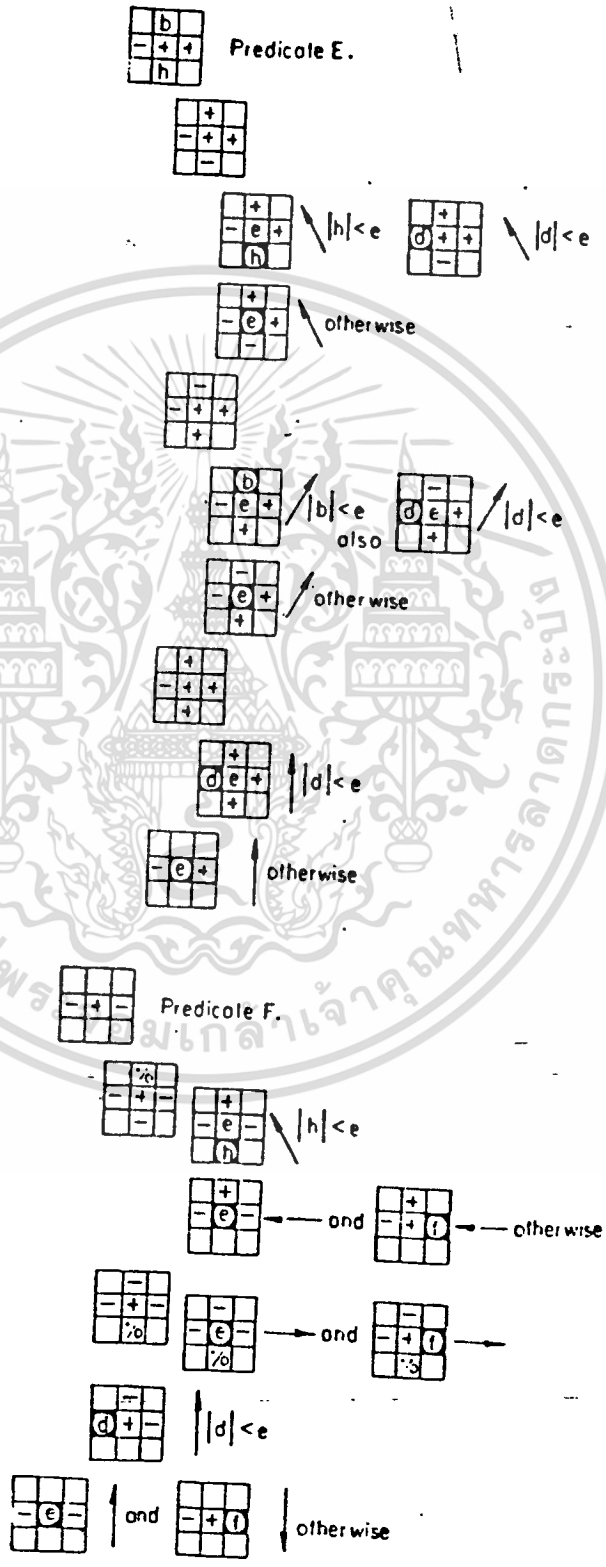
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่ม A-D



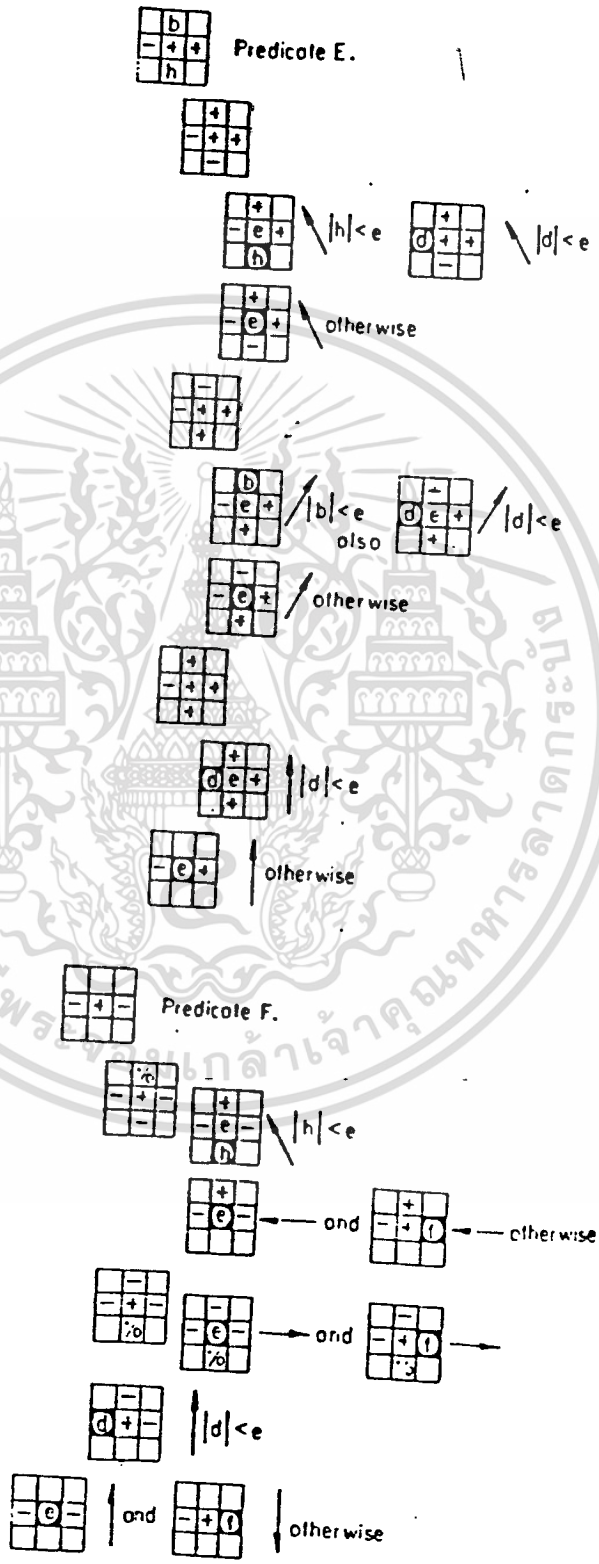
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่ม E-F



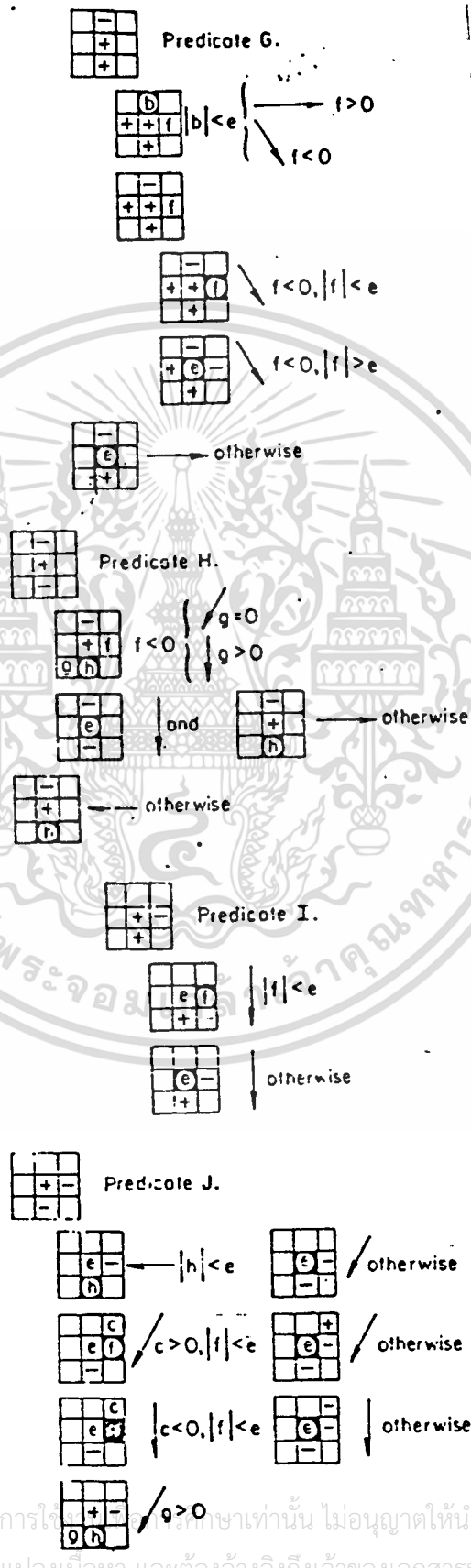
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่ม E-F



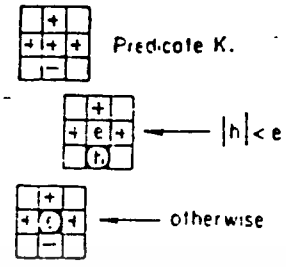
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่ม G-J



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่ม K



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้