

การศึกษาเปรียบเทียบวิธีการหาค่า จุดแบ่งแยกอัตโนมัติ

กฤษฎา วิไลลักษณ์

ธีรพันธ์ ธีรธรรมย์

ภทริยา ฐานิสโร

นักศึกษาปริญญาโท

ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น

ดร. ธนชาติ นุ่มนนท์

อาจารย์

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น

บทคัดย่อ

บทความนี้ได้นำเสนอการศึกษาเปรียบเทียบวิธีการหาค่าจุดแบ่งแยกโดยอัตโนมัติ 5 วิธี คือ วิธีของ Ostu, วิธีของ Pun, วิธีของ Kapur, วิธีของ Johannsen และ Bille, และวิธี Minimum Error, คณะผู้เขียนได้ทดลองทำการทดสอบเปรียบเทียบวิธีทั้งห้าโดยการแปลงภาพดิจิทัลระดับเทาให้เป็นภาพดิจิทัลขาวดำ จากการทดลองพบว่าวิธีการของ Ostu ให้ผลที่ดีที่สุด แต่วิธีของ Johannsen และ Bille สามารถทำงานได้รวดเร็วที่สุด

A Comparative Study of Automatic Thresholding Techniques

Krisada Wilailak

Theerapan Raruenrom

Pathreeya Thanisaro

Graduate Student

Department of Electrical Engineering

Faculty of Engineering Khon Kaen University

Dr. Thanachart Numnonda

Lecturer

Department of Computer Engineering

Faculty of Engineering Khon Kaen University

Abstract

This paper presents a comparative study of several thresholding techniques. Thresholding techniques discussed in the paper are Ostu method, Pun method, Kapur method, Johannsen and Bille method, and Minimum error method. Experiments were carried out for a comparative study. It was found that an Ostu method provides the most desirable reason, however a Johannsen and Bille method requires the least computational time.

บทนำ

การนำการประมวลผลภาพดิจิทัล (digital image processing) ไปประยุกต์ใช้งานในบางกรณี จะมีความต้องการที่จะต้องแปลงภาพดิจิทัลระดับเทา (gray scale image)¹ ให้เป็น

¹ ภาพดิจิทัลระดับเทามีระดับความเข้มสีอยู่ 256 ระดับ โดยที่ 0 คือระดับต่ำสุดจะมีสีดำ และ 255 คือระดับสูงสุดจะมีสีขาว

ภาพดิจิทัลขาวดำ (binary image)² เพื่อประโยชน์ในการวิเคราะห์ อาทิเช่น การแยกวัตถุ (bbject) ออกจากฉาก (background), และการแยกตัวอักษรเป็นต้น ซึ่งวิธีการที่นิยมใช้กันมากที่สุดในการแปลงภาพคือ วิธีการหาค่าจุดแบ่งแยก (thresholding)[1] ซึ่งวิธีนี้จะเลือกค่าจุดแบ่งแยกจากความเข้มสีระดับใดระดับหนึ่ง ระหว่าง 0 ถึง 255 ของระดับเทา มาเป็นค่าในการตัดสินใจ และกำหนดให้จุดภาพ (pixel) ในภาพดิจิทัลระดับเทาแปลงเป็นสีดำ ถ้าจุดภาพนั้นมีค่าความเข้มสีต่ำกว่าค่าจุดแบ่งแยกและแปลงเป็นสีขาวถ้าจุดภาพดังกล่าวมีค่าความเข้มสีสูงกว่าค่าจุดแบ่งแยก

วิธีการหาค่าจุดแบ่งแยกสามารถอธิบายในเชิงคณิตศาสตร์โดยอาศัยนิยามดังต่อไปนี้ กำหนดให้ภาพดิจิทัลระดับเทาที่มีจำนวนจุดภาพ $M \times N$ จุด แทนค่าด้วยเมทริกซ์ f โดยที่ $f = \{f(m,n) | m=0,1,...,M-1 \text{ และ } n = 0,1,...,N-1\}$ และให้ระดับเทาของภาพเป็นค่าเซตจำนวนเต็ม $G = \{0,1,...,l-1\}$ ($l=256$) ดังนั้นเราจะได้

$$f(m,n) \in G$$

ถ้ากำหนดให้ค่าจุดแบ่งแยกเป็นค่าระดับความเข้มสี t และให้ภาพดิจิทัลขาวดำแทนด้วยเมทริกซ์ g โดยที่ $g = \{g(m,n) | m=0,1,...,M-1 \text{ และ } n = 0,1,...,N-1\}$ ซึ่งมีค่าระดับความเข้มสีเป็นเซตของระดับความเข้มสีขาว-ดำ $B = \{b_0, b_1\} \in G$ วิธีการแปลงภาพดิจิทัลโดยวิธีการหาค่าจุดแบ่งแยกสามารถจะแสดงความสัมพันธ์ระหว่างจุดภาพของเมทริกซ์ภาพทั้งสองได้ดังนี้

$$g(m,n) = \begin{cases} b_0 & \text{if } f(m,n) \leq t \\ b_1 & \text{if } f(m,n) > t \end{cases}$$

ในปัจจุบันมีผู้นำเสนอวิธีการในการหาค่าจุดแบ่งแยกโดยอัลกอริทึมอยู่หลายวิธี[2,3] ซึ่งแต่ละวิธีมีข้อดีและข้อเสียต่างๆ กัน คณะผู้เขียนได้ทำการศึกษาเปรียบเทียบวิธีการหาค่าจุดแบ่งแยกโดยอัลกอริทึมต่างๆ ทั้งหมด 5 วิธีคือ วิธีของ Ostu[4], วิธีของ Pun[5], วิธีของ Kapur[6], วิธีของ Johannsen และ Bille[7], และวิธี Minimum Error[8] โดยมีจุดประสงค์เพื่อจะหาวิธีการหาค่าจุดแบ่งแยกโดยอัลกอริทึมที่เหมาะสมสำหรับการนำไปประยุกต์ใช้งานอื่นๆ อีกต่อไป

² ภาพดิจิทัลขาวดำจะมีระดับความเข้มสีอยู่ 2 ระดับ คือ 0 สำหรับสีดำ และ 255 สำหรับสีขาว

$$\text{เมื่อ } \mu_T = \sum_{i=0}^{t-1} i p_i$$

$$\sigma_B^2 = \omega_0 \omega_1 (\mu_1 \mu_0)^2$$

$$\text{เมื่อ } \omega_0 = \sum_{i=0}^t p_i$$

$$\omega_1 = 1 - \omega_0$$

$$\mu_1 = \frac{\mu_T - \mu_t}{1 - \omega_0}$$

$$\mu_0 = \frac{\mu_t}{\omega_0}$$

$$\mu_t = \sum_{i=0}^t i p_i$$

และ p_i คือความน่าจะเป็นของการเกิดของระดับความเข้มสี i กำหนดโดย

$$p_i = \frac{n_i}{n}$$

เมื่อ n_i เป็นจำนวนของจุดภาพที่มีระดับความเข้มสี i และ n เป็นจำนวนของจุดภาพทั้งหมด

วิธีของ Pun

วิธีการนี้นำทฤษฎีสารสนเทศ (information theory) มาใช้ในการหาค่าจุดแบ่งแยก t โดยจะพิจารณาแบ่งภาพดิจิทัลออกเป็น 2 ส่วนคือ ส่วนที่เป็นวัตถุและฉาก และค่าเอนโทรปี (entropy) ของทั้งสองส่วนสามารถคำนวณได้ดังนี้

$$H'_b = - \sum_{i=0}^t p_i \log_2 p_i$$

$$H'_w = - \sum_{i=t+1}^{I-1} p_i \log_2 p_i$$

เมื่อ H'_b และ H'_w คือค่าเอนโทรปีของวัตถุและฉากตามลำดับ

โดยที่ Pun ได้เสนอวิธีการหาค่าจุดแบ่งแยกโดยการหาค่าสูงสุดของเอนโทรปีของภาพ

$$H' = H'_b + H'_w$$

ซึ่งค่าสูงสุดดังกล่าวจะมีค่าเท่ากับการหาค่าสูงสุดของฟังก์ชัน

$$f(t) = \frac{H_t}{H_T} \frac{\log_e P_t}{\log_e \max\{p_0, \dots, p_t\}} + \left[1 - \frac{H_t}{H_T}\right] \frac{\log_e (1 - P_t)}{\log_e \max\{p_{t+1}, \dots, p_{l-1}\}}$$

เมื่อเทียบกับ t , โดยที่

$$H_t = -\sum_{i=0}^t p_i \log_e p_i$$

$$H_T = -\sum_{i=0}^{l-1} p_i \log_e p_i,$$

และ

$$P_t = \sum_{i=0}^t p_i$$

วิธีของ *Kapur, Sahoo, and Wong*

วิธีการนี้หลักการการกระจายความน่าจะเป็น (probability distribution) ของวัตถุ และฉาก โดยหากมีค่าจุดแบ่งแยกเป็น t จะมีค่าการกระจายความน่าจะเป็นดังนี้

$$\frac{p_0}{p_t}, \frac{p_1}{p_t}, \dots, \frac{p_t}{p_t}$$

และ

$$\frac{p_{t+1}}{1-p_t}, \frac{p_{t+2}}{1-p_t}, \dots, \frac{p_{l-1}}{1-p_t}$$

เมื่อ $P_t = \sum_{i=0}^t p_i$, ดังนั้นค่าเอนโทรปีสำหรับวัตถุและฉากที่คำนวณจากค่าการกระจายความน่าจะเป็น จะมีค่าดังนี้

$$H_b(t) = - \sum_{i=0}^t \frac{p_i}{P_t} \log_e \left(\frac{p_i}{P_t} \right)$$

$$H_w(t) = - \sum_{i=t+1}^{l-1} \frac{p_i}{1-P_t} \log_e \left(\frac{p_i}{1-P_t} \right)$$

และค่าจุดแบ่งแยก t^* ที่เหมาะสมก็คือค่าที่ทำให้ได้เอนโทรปีของภาพที่สูงสุด กล่าวคือ

$$t^* = \text{ArgMax} \{ H_b(t) + H_w(t) \}$$

วิธีการของ Johannsen and Bille

วิธีการนี้ใช้หลักการของเอนโทรปีของฮิสโตแกรมโดยการแบ่งระดับความเข้มของภาพออกเป็นสองส่วนเพื่อหาค่าต่ำที่สุดและเลือกค่าจุดแบ่งแยก t^* จากความสัมพันธ์

$$t^* = \text{ArgMin} \{ S(t) + \bar{S}(t) \}$$

เมื่อ

$$S(t) = \log_e \left(\sum_{i=0}^t p_i \right) - \frac{1}{\sum_{i=1}^{l-1} p_i} \left[p_t \log_e p_t + \left(\sum_{i=0}^{l-1} p_i \right) \log_e \left(\sum_{i=0}^{t-1} p_i \right) \right]$$

และ

$$\bar{S}(t) = \log_e \left(\sum_{i=t}^{l-1} p_i \right) - \frac{1}{\sum_{i=t}^{l-1} p_i} \left[p_t \log_e p_t + \left(\sum_{i=t+1}^{l-1} p_i \right) \log_e \left(\sum_{i=t+1}^{l-1} p_i \right) \right]$$

วิธีการ Minimum Error

วิธีการนี้อาศัยสมมติฐานที่ว่าฮิสโตแกรมของทั้งวัตถุและฉากต่างก็เป็นการกระจายแบบ Gaussian ดังนั้นค่าจุดแบ่งแยก t^* ที่เหมาะสมที่สุด คือค่าที่จะทำให้ฮิสโตแกรมของทั้งวัตถุและฉากมีลักษณะการกระจายใกล้เคียงกับแบบ Gaussian มากที่สุด ซึ่งสามารถคำนวณได้จากความสัมพันธ์

$$t^* = \text{ArgMin}\{J(t)\}$$

โดยที่

$$J(t) = 1 + 2\{P_1(t)\log_0 \sigma_1(t) + P_2(t)\log_0 \sigma_2(t)\} \\ - 2\{P_1(t)\log_0 P_1(t) + P_2(t)\log_0 P_2(t)\}$$

เมื่อ

$$P_1(t) = \sum_{g=0}^t h(g), \quad P_2(t) = \sum_{g=t+1}^{l-1} h(g) \\ \mu_1(t) = \frac{\sum_{g=0}^t h(g)g}{P_1(t)}, \quad \mu_2(t) = \frac{\sum_{g=t+1}^{l-1} h(g)g}{P_2(t)}$$

$$\sigma_1^2(t) = \frac{\sum_{g=0}^t (g - \mu_1(t))^2 h(g)}{P_1(t)}$$

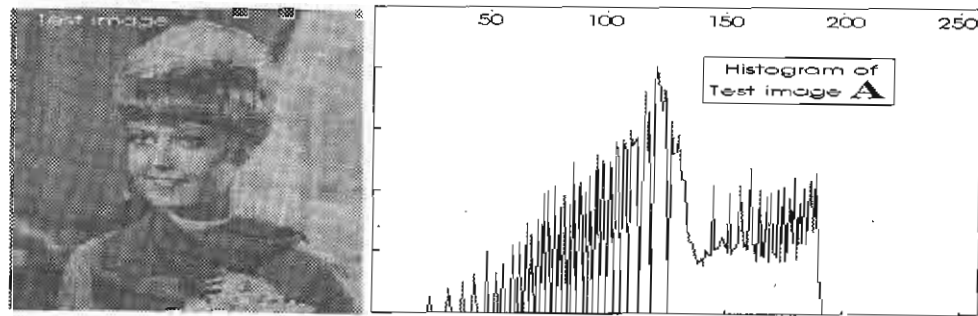
$$\sigma_2^2(t) = \frac{\sum_{g=t+1}^{l-1} (g - \mu_2(t))^2 h(g)}{P_2(t)}$$

ผลการทดลอง

คณะผู้เขียนได้พัฒนาโปรแกรมบนเครื่องไมโครคอมพิวเตอร์โดยใช้โปรแกรม MATLAB และภาษา C เพื่อการศึกษาเปรียบเทียบความสามารถในการเลือกค่าจุดแบ่งแยกอัตโนมัติโดยใช้วิธีการทั้ง 5 วิธีที่กล่าวผ่านได้ วิธีการทดสอบคือ การนำภาพดิจิตอลระดับเทาสองภาพที่มีขนาด 256 x 256 จุดภาพมาแปลงเป็นภาพดิจิตอลขาวดำ โดยในรูปที่ 1 และ 2

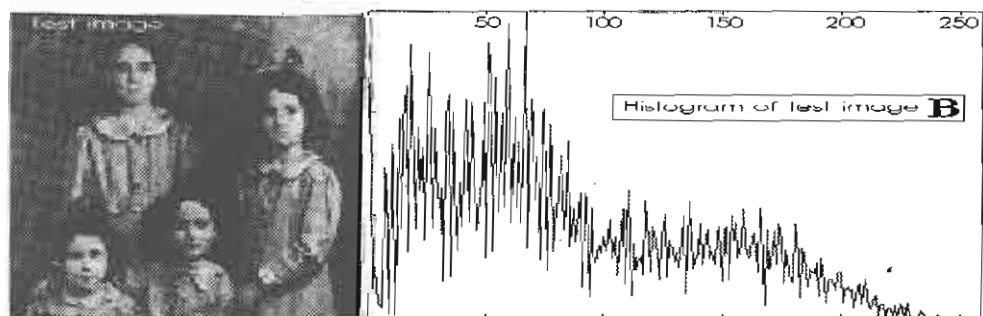


ได้แสดงภาพทดสอบทั้งสอง (เรียกว่า ภาพทดสอบ A และ ภาพทดสอบ B) พร้อมทั้งฮิสโตแกรมของภาพตามลำดับ ส่วนผลการแปลงภาพเป็นภาพดิจิทัลขาวดำสำหรับภาพทดสอบทั้งสองภาพนั้น ได้แสดงไว้ในรูปที่ 3 และ 4 ตามลำดับ



(a) (b)

รูปที่ 1 (a) ภาพทดสอบ A (b) ฮิสโตแกรมภาพ



(a) (b)

รูปที่ 2 (a) ภาพทดสอบ B (b) ฮิสโตแกรมภาพ



(a)



(b)



(c)



(d)



(e)

รูปที่ 3 ผลการแปลงภาพทดสอบ A โดยวิธี (c) วิธีของ Kapur,

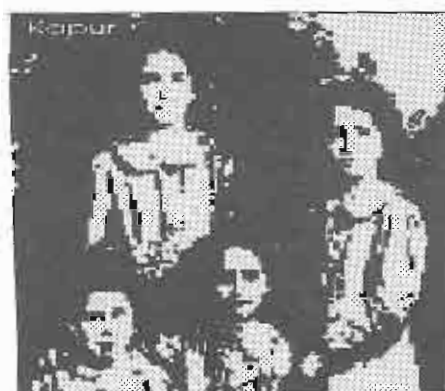
(d) วิธีของ Johannsen และ Bille, และ (e)วิธี Minimum Error



(a)



(b)



(c)



(d)



(e)

รูปที่ 4 ผลการแปลงภาพทดสอบ B โดยวิธี (a) วิธีของ Ostu, (b) วิธีของ Pun, (c) วิธีของ Kapur, (d) วิธีของ Johannsen และ Bille, และ (e)วิธี Minimum Error

จากผลการทดสอบจะเห็นว่าเป็นเรื่องยากที่จะสรุปได้ว่าวิธีใดเหมาะสมที่สุดหากเพียงสังเกตจากรูปภาพ ดังนั้นเพื่อให้การตัดสินใจว่าวิธีใดเหมาะสมที่สุดนั้น คณะผู้เขียนจึงได้คำนวณการเป็นรูปแบบของภาพ (uniformity measure) ซึ่งสามารถคำนวณได้จากสมการต่อไปนี้[9]

$$U(t) = 1 - \frac{\sigma_1^2 + \sigma_2^2}{C_1}$$

และค่า $U(t)$ เปรียบเทียบวิธีการทั้ง 5 สำหรับภาพทดสอบทั้งสองแสดงไว้ในตารางที่ 1 นอกจากนี้เวลาที่ใช้ในการคำนวณหาค่าจุดแบ่งแยกสำหรับแต่ละวิธีได้แสดงไว้ในตารางที่ 2

ตารางที่ 1

ค่าจุดแบ่งแยกและค่า $U(t)$ ที่จากวิธีต่างๆ

วิธี	ภาพทดสอบ A		ภาพทดสอบ B	
	ค่า t	$U(t)$	ค่า t	$U(t)$
Ostu	135	1.000	100	1.000
Pun	106	0.978	50	0.541
Kapur	133	0.998	111	0.974
Johannsen	125	0.980	72	0.929
Min error	120	0.981	80	0.997

ตารางที่ 2

เวลาที่ใช้ในการคำนวณของแต่ละวิธี

วิธี	ภาพ A (นาทีก)	ภาพ B (นาทีก)
Ostu	0.16	0.22
Pun	8.02	3.95
Kapur	6.15	3.08
Johannsen	0.06	0.05
Min error	5.11	3.84

จากฮิสโตแกรมของภาพทดสอบทั้งสองจะเห็นว่า ภาพทดสอบ A จะสามารถหาจุดแบ่งแยกได้ง่ายกว่าภาพทดสอบ B เพราะฮิสโตแกรมของภาพทดสอบ A สามารถแสดงให้เห็นการกระจายของระดับความเข้มสีสามารถแบ่งเป็นส่วนของวัตถุและฉากได้เด่นชัดกว่า ซึ่งจากผลการทดลองจะเห็นว่าวิธีการทั้งห้าได้ค่าจุดแบ่งแยกที่ใกล้เคียงกัน ซึ่งเมื่อพิจารณาจากภาพดิจิทัลขาวดำที่แปลงมาได้หรือค่า $U(t)$ จะเห็นว่าได้ผลที่ใกล้เคียงกัน แต่สำหรับภาพทดสอบ B ซึ่งมีความซับซ้อนของฮิสโตแกรมมากกว่าจะเห็นว่า วิธีการ Ostu ให้ค่าที่ดีที่สุดเมื่อเทียบกับวิธีอื่น เมื่อประเมินจากค่า $U(t)$ แต่กระนั้นก็ตามเมื่อประเมินด้วยสายตาภาพดิจิทัลขาวดำที่แปลงมาไม่ได้แตกต่างกันมากนัก ยกเว้นวิธีของ Pun ที่ได้ผลไม่ดีนัก

ถ้าพิจารณาจากความรวดเร็วในการคำนวณของแต่ละวิธีจะได้ว่าวิธีการของ Johannsen และ Bille ใช้เวลาในการคำนวณน้อยที่สุด โดยที่วิธี Ostu ใช้เวลาน้อยเป็นอันดับสองแต่ก็ไม่ได้แตกต่างกับเวลาของวิธีการของ Johannsen และ Bille มากนัก ส่วนวิธีอื่นๆ ต่างใช้เวลาในการคำนวณมากกว่าทั้งสองวิธีนี้มาก

สรุป

บทความนี้ได้นำเสนอวิธีการหาค่าจุดแบ่งแยกเพื่อแปลงภาพดิจิทัลระดับเทาให้เป็นภาพดิจิทัลขาวดำ โดยที่คณะผู้เขียนได้ศึกษาทดลองนำวิธีการที่ใช้อยู่ปัจจุบัน 5 วิธีมาเปรียบเทียบกัน ในบทความได้เสนอให้เห็นตัวอย่างผลการทดลองจากภาพทดสอบ 2 ภาพ และแสดงให้เห็นว่าวิธีการที่เหมาะสมจากทั้ง 5 วิธีนี้คือวิธีของ Ostu

เอกสารอ้างอิง

- [1] R.C. Gonzalez and R.E.Woods, "*Digital Image Processing*", Addison-Wesley, Massachusetts, 1993.
- [2] T. Numnonda, "*Fillet Defect Detection Using Machine Vision*", Tech. Report, School of engineering, University of Auckland, New Zealand, 1990.

- [3] P.K.Sahoo, S. Soltani, A.K.C.Wong, and Y.C.Chen, "**A Survey of Thresholding Techniques**", Comp. Vision, Graph. and Image Proc., Vol. 41, pp. 233-260, 1988.
- [4] N. Ostu, "**A Thresholding Selection Method from Gray-level Histograms**", IEEE Trans. Sys., Man and Cybern., Vol. 9, pp. 62-66, 1979.
- [5] T. Pun, "**A New Method for Gray-level Picture Thresholding Using the Entropy of the Histogram**", Signal Processing, Vol. 2, pp. 223-237, 1980.
- [6] J.N. Kapur, P.K.Sahoo, and A.K.C. Wong, "**A New Method for for Gray-level Picture Thresholding Using the Entropy of the Histogram**", Comp. Vision, Graph. and Image Proc., Vol. 29, pp. 273-285, 1985.
- [7] G. Johannsen and J. Bille, "**A Threshold Selection Method Using Information Measures**", in Proc. of the 6th Int. Conf. Pattern Recognition, Germany, pp. 140-143, 1982.
- [8] J. Kittle and J. Illingworth, "**Minimum Error Thresholding**", Pattern Recognition, Vol. 19, pp. 41-47, 1986.
- [9] M.D. Levine and A.M.Nazif, "**Dynamic Measurement of Computer Genereated Image Segmentations**", IEEE Trans. Patt. Anal. and Mach. Vision, Vol. 7, pp. 155-164, 1985.