

## BAB II

### TINJAUAN PUSTAKA

Perkembangan keilmuan dibidang *computer vision* dan *image processing* beberapa dekade terakhir, yang ditandai dengan banyaknya penelitian berkaitan dengan bidang tersebut; berbagai algoritma dan metode tercipta banyak metode untuk memecahkan masalah pada *computer vision* dan *image processing*. Salah satu masalah mendasar pada *computer vision* dan *image processing*, yang pemecahan-nya terus berkembang adalah *image segmentation*. Dari banyaknya teknik dan algoritma yang ada untuk memecahkan masalah tersebut; *level set method* (LSM) adalah salah satu teknik yang populer di dunia penelitian untuk melakukan *image segmentation* (Wen, et al., 2015). Penelitian yang dilakukan, membuat LSM memiliki beragam kombinasi dalam menyelesaikan masalah segmentasi tertentu.

Pertama kali LSM digunakan oleh Osher dan Sethian untuk menangkap (*capturing*) tulisan bergerak pada tahun 1987. Kemudian membuat LSM menjadi populer dan berdampak besar pada beragam aplikasi, seperti: *computational geometry*, *fluid dynamics*, *image processing* dan *computer vision*. Namun, memiliki masalah dalam proses *re-initialization*, yang menyebabkan beragam kesalahan dan hasil yang tidak stabil (Li, et al., 2010) (Zhang, et al., 2013). Kemudian lahir metode baru bernama *Distance Regularized Level Set Evolution* (DRLSE), yang diciptakan untuk mengatasi masalah tersebut.

Banyak peneliti yang mengusulkan metode DRLSE ini, salah satunya adalah Chunming Li, et al.. Pada penelitiannya, berusaha menghilangkan proses *re-initialization* pada LSM dengan menggunakan *distance regularization*. Selain itu, metode ini juga berhasil mempertahankan kestabilan hasil segmentasi. Metode yang diusulkan ini, juga dapat diterapkan secara sederhana dan memiliki *numerical scheme* yang efisien dibanding dengan yang *conventional* LSM. Model *active contour* pada metode ini juga dapat menggunakan *large time step* untuk mengurangi jumlah iterasi yang perlu dilakukan.

Selain metode DRLSE, terdapat metode LSM bernama *reaction-diffusion* (RD) yang diusulkan oleh Kaihua Zhang et al.. Pada penelitiannya juga, berusaha menghilangkan proses *re-initialization* pada LSM dengan cara yang berbeda. Pada metode ini, Zhang menggabungkan keuntungan dari *phase transition* pada metode RD dan LSM; yang memebentuk RD-LSE *equation*. Selain menghilangkan proses *re-initialization*, metode ini juga dapat digeneralisasi menjadi kerangka terpadu, yang mana *level set evolution* (LSE) dapat berbasis PDE atau *variational*. Kelebihan lainnya adalah metode ini memiliki *boundary anti-leakage* yang tinggi dan *anti-noise* dibanding dengan *state-of-the-art method*. Pada peneltian ini, Zhang membandingkan waktu eksekusi metodenya dengan beberapa metode yang dibuatnya. Tabel 2.1 dan Tabel 2.2 berikut adalah tabel perbandingan waktu eksekusi dengan menggunakan CPU yang dilakukan oleh Zhang dalam jurnal yang berjudul '*Reinitialization-Free Level Set Evolution*'.

Tabel 2.1 Perbandingan Time CPU dan Iterasi (Iter) antara Metode RD, GDRGLSE1, GDRGLSE2, dan GDRGLSE3

Metode	Image 1		Image 2		Image 3	
	Size : 100 x 100 pixels					
	Edge-based model with $\delta_{l,p}$					
	Time(s)	Iter	Time(s)	Iter	Time(s)	Iter
RD	<b>4,8</b>	<b>500</b>	<b>5,0</b>	<b>520</b>	<b>5,8</b>	600
GDRGLSE1	6,2	550	6,5	580	6,7	<b>590</b>
GDRGLSE1	9,3	630	9,7	660	10,1	690
GDRGLSE3	14,2	1000	17,0	1200	17,8	1250

Tabel 2.2 Perbandingan Time CPU dan Iterasi (Iter) antara Metode RD, GDRGLSE1, GDRGLSE2, dan GDRGLSE3

Metode	Image 4		Image 5		Image 6	
	Size : 96 x 101 pixels					
	CV model					
	Time(s)	Iter	Time(s)	Iter	Time(s)	Iter
RD	4,8	150	5,1	<b>160</b>	<b>5,1</b>	<b>160</b>
GDRGLSE1	5,1	150	9,3	200	10,9	230
GDRGLSE1	4,7	<b>130</b>	11,4	200	12,9	230
GDRGLSE3	<b>4,0</b>	<b>130</b>	18,1	210	68,9	800

Beberapa tahun terakhir ini, banyak peneliti yang menargetkan implementasi dari LSM menggunakan *parallel architectures*, yang melibatkan GPU (Gutierrez, et al., 2016). Kemudian, pada tahun 2000-an LSM mulai berjalan pada GPU. Penerapan GPU pada LSM ini, pertama kali dilakukan oleh Rumpf dan Strzodka. Setelah dirilisnya bahasa pemrograman CUDA C pada tahun 2007, performa yang lebih tinggi berhasil dicapai (Güler & Çınar, 2013).

Pada penelitian ini, akan menerapkan *parallel programming* pada metode LSM yang bernama GDRLSE3 yang dibuat oleh Kaihua Zhang dalam penelitiannya. Paralelisasi akan dilakukan dengan menggunakan GPU dan bahasa pemrograman CUDA C, dengan harapan dapat mempercepat waktu komputasi yang dilakukan tanpa mengubah hasil akhir proses segmentasi.