# BAB II

# TINJAUAN PUSTAKA

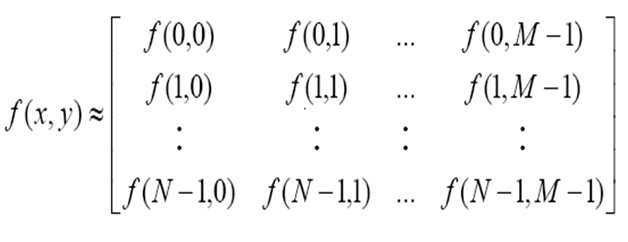
Bab ini berisi uraian mengenai penelitian terkait yang pernah dilakukan sebelumnya mengenai pengenalan objek menggunakan ekstraksi *region adjacency* *graph* dan pencocokan dengan *graph matching*. Pada bagian ini juga dipaparkan mengenai teori serta tulisan yang berkaitan dengan penelitian yang dilakukan.

**2.1 Landasan Teori**

Berikut adalah beberapa landasan teori yang digunakan sebagai dasar-dasar pemikiran dalam penelitian ini.

### 2.1.1 Citra Digital

Citra digital diartikan sebagai suatu fungsi intensitas cahaya dua dimensi, yang dinyatakan oleh , di mana nilai pada koordinat spasial menyatakan intensitas citra pada titik tersebut. Citra sebagai salah satu komponen multimedia memegang peranan sangat penting sebagai bentuk informasi visual. Meskipun sebuah citra kaya informasi, namun seringkali citra yang kita miliki mengalami penurunan mutu sehingga menjadi lebih sulit diinterpretasi karena informasi yang disampaikan oleh citra tersebut menjadi berkurang (Munir, 2004). Gambar 2.1 menunjukkan representasi matriks untuk citra digital.



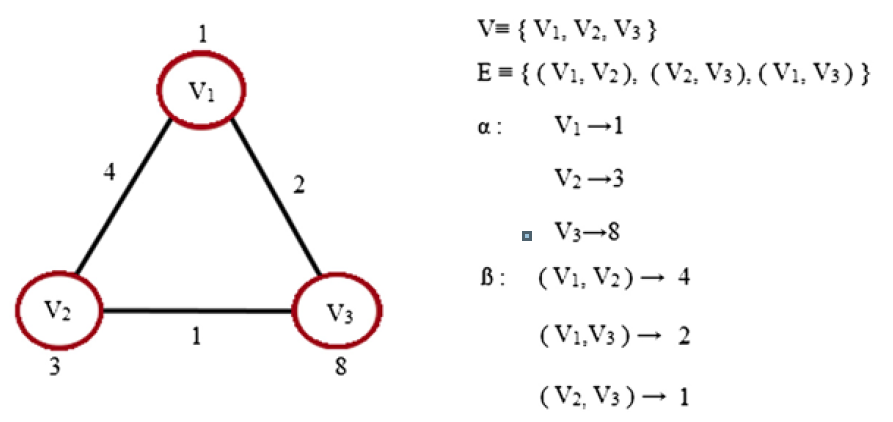
Gambar 2.1 Representasi Matriks untuk Citra Digital

### 2.1.2 Teori Dasar *Graph*

*Graph* adalah sebuah himpunan yang memiliki sejumlah titik-titik, yang disebut simpul (*vertex* atau *node*), yang saling dihubungkan melalui garis-garis yang disebut sisi (*edge*) (Sharma, et al., 2012). Suatu sisi dapat menghubungkan suatu simpul dengan simpul yang sama, sisi tersebut dinamakan *loop*.

Secara umum *graph* G didefinisikan sebagai pasangan himpunan (*V,E*), ditulis dengan yang dalam hal ini *V* adalah himpunan tidak kosong dari simpul-simpul (*nodes*) dan adalah himpunan sisi (*edges*) yang menghubungkan sepasang simpul (Angelia, 2011).

Sebuah *graph* dikatakan *Attributed Relational Graph* (ARG) ketika simpul-simpul dan sisi-sisinya direpresentasikan dalam atribut-atribut tertentu. Atribut simpul untuk simpul dinotasikan sebagai sebuah vektor , di mana adalah jumlah dari atribut-atribut simpul dalam vektor , dan atribut-atribut sisi (*weights*) untuk sisi dinotasikan sebagai , di mana adalah jumlah atribut-atribut sisi dalam vektor (Sharma, et al., 2012). Gambar 2.2 menunjukkan contoh ARG.



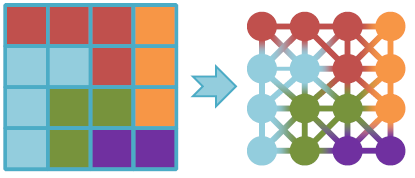
Gambar 2.2 Contoh *Attributed Relational Graph*

### 2.1.3 Segmentasi Berbasis *Graph*

Sebelum dibangun sebuah RAG diperlukan pengelompokkan piksel-piksel ke dalam kelompok-kelompok (*regions)* tertentu, proses ini dinamakan segmentasi. Salah satu metode segmentasi citra berbasis *graph* yaitu metode yang diusulkan oleh Felzenzswalb.

Metode segmentasi ini diusulkan pada *International Journal of Computer Vision* (2004) ke-59 dengan judul “*Efficient Graph-Based Image Segmentation*” yang ditulis oleh Pedro F. Felzenszwalb. Metode ini menggunakan *graph* dalam bentuk *Minimum Spanning Tree* (MST) dan pendekatan algoritma *Kruskal* sebagai representasi untuk proses segmentasi. Algoritma ini telah diujikan untuk citra real maupun sintetis (buatan). *Running time* untuk algoritma ini berjalan secara linier dengan jumlah *edges* (Felzenszwalb, 2004). Metode ini dapat mengukur batas-batas regional dengan membandingkan dua buah kuantitas: berdasarkan perbedaan intensitas batas regional yang bersebrangan, dan berdasarkan intensitas antar piksel-piksel yang bertetanggaan di setiap regional.

Melalui pendekatan *graph*. Dapat didefinisikan sebagai *graph* tidak berarah dengan simpul , dan sisi menunjukkan hubungan sepasang piksel yang bertetanggan. Dengan kata lain merupakan piksel-piksel dalam sebuah citra dan bobot dari sebuah sisi yang diukur berdasar intensitas perbedaan sepasang piksel . Contoh penggambaran *graph* untuk citra digital ditunjukkan oleh Gambar 2.3.



Gambar 2.3 Penggambaran citra dengan pendekatan graph

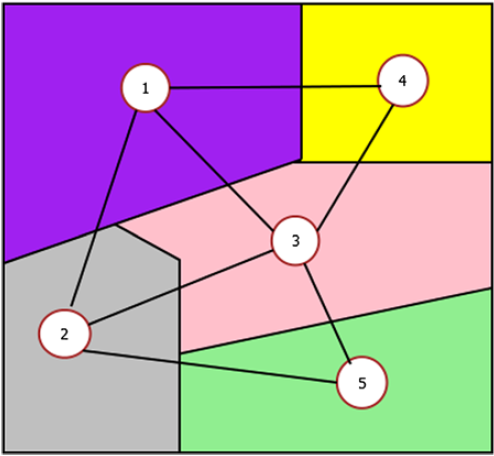
Algoritma pada metode ini berisi masukan berupa *graph* , dengan simpul dan sisi. Keluaran yang akan dihasilkan adalah segmentasi ke dalam komponen yang merupakan gabungan komponen-komponen regional yang telah dikelompokkan. Langkah dasar *psudo code* algoritma:

|  |
| --- |
| for all edges from w\_min to w\_max  if ( edge is internal to region )  continue  else  compute Int(C1), Int(C2), D(C1,C2)  if ( boundary exists between C1, C2)  continue  else  merge C1, C2 into new region |

### 2.1.4 Region Adjacency Graph Mean Color

*Region Adjacency Graph* (RAG) merupakan sebuah ARG yang memiliki simpul-simpul yang dapat merepresentasikan kumpulan daerah (*regions*) dan sisi-sisi yang dapat merepresentasikan hubungan antar simpul yang berdekatan. RAG memberikan keefektifan dalam aplikasi untuk representasi informasi dari suatu citra. RAG telah banyak digunakan dalam bidang segmentasi citra berwarna (Tremeau A, 2000).

*Region Adjacency Graph* (RAG) menghubungkan bagian-bagian citra yang telah terpartisi melalui proses sebelumnya. Manfaat utama dari RAG yaitu dapat menunjukkan “*spatial view*” dari citra. Salah satu cara untuk merepresentasikan RAG yang berisi kumpulan simpul dari setiap partisi. Penggambaran RAG ditunjukkan Gambar 2.4.

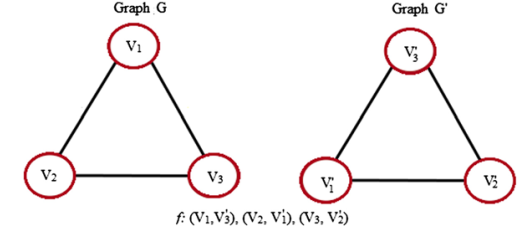


Gambar 2.4 Penggambaran Region Adjacency Graph

### 2.1.5 Graph Matching

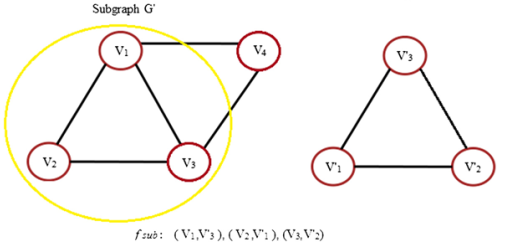
*Graph Matching* adalah proses pembandingan dua buah *graph* untuk mengukur sebuah hubungan kemiripan maupun ketidakmiripan antar simpul dan sisi dari kedua *graph* tersebut. Hal ini mengacu pada proses pemetaan dari simpul-simpul suatu *graph* ke simpul-simpul dari *graph* lain yang memenuhi batas-batas atau kriteria optimal. Berikut merupakan konsep dasar dari proses metode *graph matching*:

1. *Graph Isomorphism*. Proses ini menemukan struktur yang tepat antar dua buah objek *graphs*. Di dalamnya terjadi korespondensi satu-satu antar simpul dan sisi dari kedua *graph*. Gambar 2.5 menunjukkan contoh *Graph Isomorphism*.



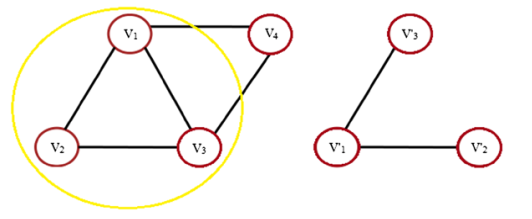
Gambar 2.5 Graph Isomorphism

1. *Subgraph Isomorphism*. Proses ini menemukan sebuah *subgraph*  dalam sebuah *graph* . Jika simpul-simpul yang berhubungan di dalam graph dihapus, akan ditemukan sebuah *subgraph* . Gambar 2.6 menunjukkan contoh *Subgraph Isomorphism*.



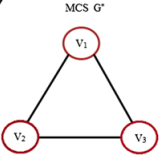
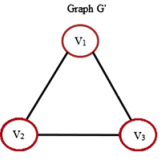
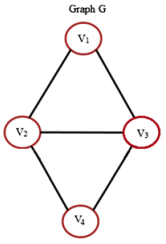
Gambar 2.6 Subgraph Isomorphism

1. *Monomorphism*. Proses ini lebih fleksibel dari *subgraph isomorphism* karena disini diperbolehkan ada sisi lain pada simpul di dalam *graph* yang lebih besar. Gambar 2.7 menunjukkan contoh *monomorphism*.



Gambar 2.7 Monomorphism

1. *Maximum Common Subgraph* (MCS). Sebuah MCS dari dua *graphs*, dan , adalah sebuah *graph* yang merupakan bagian dari dan . Sehingga MCS memiliki jumlah simpul maksimal yang mungkin pada kedua *graphs.* Gambar 2.8 menunjukkan contoh MCS.



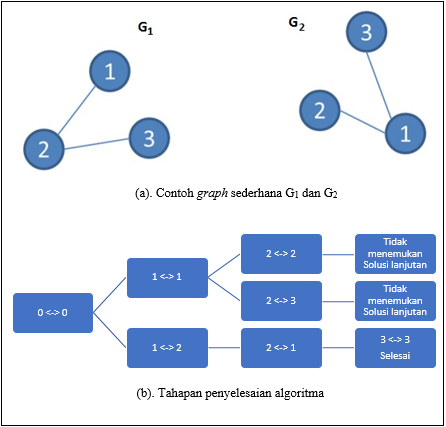
Gambar 2.8 Maximum Common Subgraph

*Graph Matching* telah menjadi topik penelitian bidang *Computer Science* selama beberapa dekade (Riesen, 2015). Dua kategori utama dalam *graph matching* yaitu *exact graph matching* dan *inexact graph matching* (atau *error-tolerant graph matching*).

1. *Exact Graph Matching.* Kategori ini bertujuan untuk menemukan hubungan yang identik antara dua objek *graphs* tanpa mengubah simpul atau sisi dari sebuah *graph*. Yang artinya terdapat hubungan *bijective* antar simpul dari dan .
2. *Inexact Graph Matching* atau *Error-Tolerant Graph Matching.* Di samping itu, kategori ini bekerja untuk mengukur seberapa mirip atau seberapa berbeda sebuah *graph* dengan *graph* lainnya. Hal ini dilakukan ketika *exact graph matching* tidak dapat dilakukan, karena adanya perbedaan jumlah simpul dan sisi pada bagian *graph* yang lain.

### 2.1.6 *VF2 Algorithm*

Algoritma ini diperkenalkan oleh Cordella pada tahun 2001. Algoritma ini dapat digunakan untuk masalah *graph* isomorphismdan *graph-subgraph* isomorphism yang mana merupakan bagian dari *exact graph matching* (Cordella, Foggia, & Vento, 2001). VF2 merupakan perkembangan dari algoritma Ullman yang sebelumnya hanya bisa digunakan untuk masalah *graph isomorphism*.

Secara sederhana, algoritma ini bekerja dengan melakukan pemetaan dari simpul pada ke setiap simpul-simpul di . Gambar 2.9 menunjukkan contoh sederhana penerapan algoritma ini.

Gambar 2. 9 Contoh sederhana penerapan algoritma VF2

### *Graph Edit Distance*

*Graph edit distance* (GED) adalah sebuah teknik untuk mengukur kemiripan antar dua buah *graph*. Konsep awal teknik ini pertama dikenalkan oleh Alberto Sanfeliu dan King-Sun Fu pada tahun 1983. Aplikasi utamanya yaitu pada *inexact graph matching* atau *error-tolerant* *pattern recognition* dalam *machine learning*.

GED antar dua buah *graph* dan secara matematika dituliskan sebagai , dan dapat didefinisikan sebagai berikut:

Dimana dinotasikan sebagai sebuah set *edit paths* yang mentransformasikan ke bentuk isomorfis dari dan merupakan nilai biaya dari setiap operasi perubahan *graph*.

Beberapa operasi dasar pada *graph edit distance* meliputi:

* *Vertex Insertion*, menambahkan sebuah simpul beratribut pada sebuah *graph*.
* *Vertex Deletion*, menghapus sebuah simpul dari sebuah *graph*.
* *Vertex Subtitution*, mengganti nilai atribut dari sebuah simpul.
* *Edge Insertion*, memberikan sisi baru yang telah diberi nilai bobot pada sepasang simpul.
* *Edge Deletion*, menghapus sisi antar sepasang simpul.
* *Edge Substitution*, mengganti nilai bobot pada sebuah sisi.

**2.2 Penelitian Sebelumnya**

Berikut adalah beberapa penelitian sebelumnya yang berkaitan dengan penelitian ini.

1. ***Determining Similiarity in Histological Images using Graph-Theoretic Description and Matching Methods for Content-Based Image Retrieval in Medical Diagnostics*** (Sharma, et al., 2012)

Penelitian ini mendeskripsikan sebuah metode untuk menentukan kemiripan antar citra-citra jaringan makhluk hidup berbasis teori deskripsi *graph*, dengan tujuan *Content Based Retrieval*. Pencapaian tertinggi dari metode ini telah didapatkan pada representasi *graph* pada gambar biopsi payudara dan pencarian *tree* berbasis teknik *inexact graph matching* telah digunakan dalam memfasilitasi proses *automatic retrieval* dari citra yang secara struktural mirip dengan gambar yang diberikan dari *database* besar.

Hasil dari evaluasi yang dilakukan menunjukkan efektifitas dan superioritas *Graph-Based Image Retrieval* melebihi teknik berbasis histogram. Kompleksitas *graph matching* yang digunakan telah dikurangi berdasarkan *state of the art* pada metode *inexact matching* yang optimal dengan menerapkan pra-syarat untuk pencocokan simpul dan desain yang canggih dari fungsi estimasi, terutama fungsi prognosis. Tabel 2.1 menunjukkan hasil evaluasi *precision* untuk metode berbasis teori graph dalam penelitian ini.

Tabel 2.1 Tabel precision untuk metode berbasis teori graph

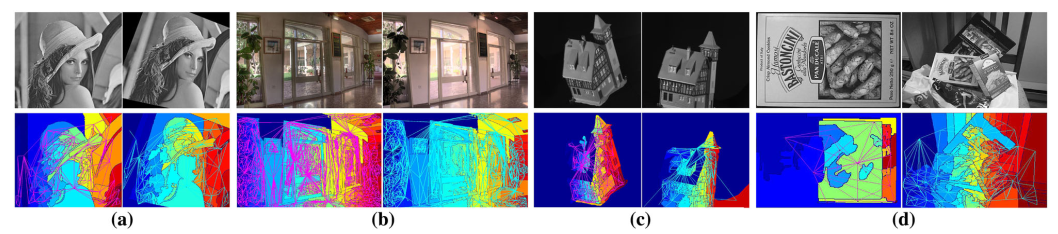
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Ps* / Window Size | 64 x 64 | 128 x 128 | 256 x 256 | 512 x 512 | **Average *Ps*** |
| *P10* | 80 | 55 | 63 | 70 | **67** |
| *P20* | 63 | 44 | 53 | 40 | **50** |
| *P30* | 58 | 39 | 50 | 36 | **46** |
| *P40* | 53 | 33 | 38 | 33 | **39** |
| *P50* | 46 | 29 | 35 | 29 | **35** |

Pada tabel tersebut dapat dilihat bahwa nilai *precision* tertinggi untuk metode ini berada pada citra 64 x 64 yaitu senilai **80 %**. Adapun pada setiap *window size* nilai tertinggi selalu terjadi pada citra 64 x 64. Dan rata-rata setiap ukuran piksel terbesar terdapat pada *windows size* ukuran 10.

1. ***Matching Images based on Consistency Graph and Region Adjacency Graphs*** (Luo, Zhou, Xu, & Zhang, 2017)

Metode pencocokan citra berdasarkan *regions* memberikan keunggulan yang lebih dibanding teknik pencocokan titik/piksel. Penelitian ini mengusulkan sebuah algoritma baru dalam pencocokan *region* citra berdasarkan konsistensi *graph* dan *region adjacency graphs*.

Eksperimen ini menghasilkan *benchmark* untuk menunjukkan bahwa metode yang diusulkan dapat menemukan secara cepat hubungan yang sesuai antar citra dengan penerangan, rotasi, dan transformasi yang berbeda. Metode yang diusulkan bekerja walau dengan beberapa perubahan rotasi, penerangan, dan transformasi. Ketika kebanyakan metode segmentasi hanya konsisten dengan *over-segmentation* dan *under-segmentation*, metode ini dapat menemukan dengan benar hubungan yang sesuai. Gambar 2.10 menunjukkan beberapa citra *bencmark* berikut hasil ekstraksinya dalam *region adjacency graph*.

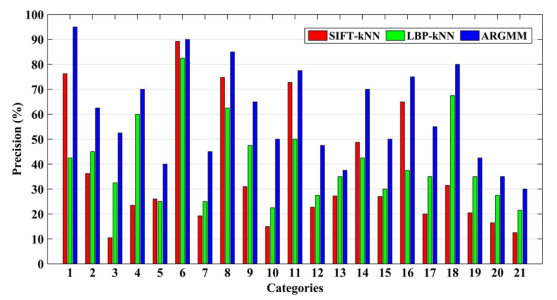


Gambar 2.10 Benchmark dan hasil representasi region adjacency graph

### Region-Based Retrieval of Remote Sensing Images Using an Unsupervised Graph-Theoretic Approach (Chaudhuri, Demir, & Bruzzone, 2016)

Jurnal ini memperkenalkan sebuah pendekatan *unsupervised* baru dalam *region-based retrieval* pada citra penginderaan jauh (*remote sensing images*). Pendekatan yang diusulkan yaitu melalui 2 tahap: 1) memodelkan setiap citra dengan sebuah *graph*, yang menyediakan representasi citra berbasis *region* yang mengombinasikan informasi lokal dan spasial, dan 2) mengambil citra dalam arsip yang memiliki kemiripan terbesar dengan citra masukan dengan mengevaluasi kemiripan berdasar *graph*.

Penelitian yang dilakukan pada arsip citra udara menunjukkan bahwa pendekatan yang diusulkan secara signifikan meningkatkan kinerja proses *retrieval* dibanding *state-of-the-art* metode *unsupervised RS image retrieval*. Gambar 2.11 menunjukkan hasil *precision* dari rata-rata percobaan (dalam 20 percobaan per kategori) untuk seluruh kategori yang diujikan dalam penelitian ini menggunakan SIFT-kNN, LBP-kNN, dan ARGMM yang diusulkan.



Gambar 2.11 Rata-rata precision pada percobaan setiap kategori citra

Dari gambar di atas dapat dilihat, bahwa nilai rata-rata *precision* hasil klasifikasi menggunakan metode yang diusulkan selalu lebih baik dibanding metode lainnya. Dimana pada citra kategori ke-1 metode ini menghasilkan nilai tertinggi yaitu *precision* > 90 %.