**PENGENALAN OBJEK PADA CITRA DIGITAL BERBASIS *GRAPH* MELALUI PENENTUAN KEMIRIPAN MENGGUNAKAN METODE *GRAPH MATCHING***

SKRIPSI

Diajukan untuk menempuh ujian sarjana

pada Fakultas Matematika dan Ilmu Pengetahuan Alam

Universitas Padjadjaran

**MUHAMAD WIJAYA ADISAPUTRA  
NPM 140810140034**



**UNIVERSITAS PADJADJARAN**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**PROGRAM STUDI S-1 TEKNIK INFORMATIKA**

**JATINANGOR**

**201**

# BAB I

# PENDAHULUAN

Bab ini menguraikan tentang latar belakang penelitian, identifikasi masalah, batasan masalah, maksud dan tujuan penelitian, kegunaan penelitian, metodologi penelitian, dan sistematika penulisan.

## Latar Belakang

Pengenalan pola atau *pattern recognition* merupakan kemampuan mengenali objek berdasarkan ciri-ciri dan pengetahuan yang diamati dari objek-objek tersebut. Implementasi pengenalan pola salah satunya yaitu terhadap citra digital. Pengenalan pola pada citra digital telah diimplementasikan di berbagai area kehidupan, seperti pada pengenalan wajah, huruf, sidik jari, dan berbagai objek lainnya. Saat ini, bidang tersebut memiliki berbagai macam permasalah yang masih sukar dipecahkan, sehingga masih gencar dikembangkan oleh para peneliti. Pada dasarnya, proses ini bekerja dengan mencari kemiripan fitur-fitur suatu citra digital dengan citra digital lainnya.

Untuk memperoleh fitur-fitur pada setiap citra digital perlu dilakukan proses ekstraksi fitur atau *feature extraction*. Fitur-fitur ini akan digunakan sebagai parameter khusus untuk membedakan suatu objek citra dengan citra lainnya. Sebelumnya, ekstraksi fitur yang biasa digunakan pada pengenalan citra adalah dengan menggunakan ekstraksi ciri berupa bentuk, warna, ukuran, geometri, tekstur, dan lain-lain.

Teknik analisis citra berbasis piksel yang tradisional tidak menghasilkan ekstraksi yang efisien karena hanya merepresentasikan konten. Pendekatan yang sangat menjanjikan adalah dengan mengekstraksi *graph* dari citra (Akmal, Suwardi, & Munir, 2017). Hal ini dikarenakan *graph* diyakini memiliki struktur penjelas yang lebih efektif berkat kemampuannya untuk merepresentasikan informasi relasional. Selain itu, *graph* juga dapat digunakan untuk menyediakan deskripsi citra yang lebih efisien melalui kumpulan simpul dengan atribut ditentukan menurut komponen citra, serta *edges* dengan pendekatan nilai bobot yang sesuai dengan kebutuhan citra (Sharma, et al., 2012). Pada beberapa penelitian telah dilakukan uji coba menggunakan ekstraksi fitur berbasis *graph*, di mana penelitian-penelitian tersebut menghasilkan fitur yang lebih fleksibel dan kuat.

*Graph matching* merupakan sebuah teknik yang *powerful* dan *robust* untuk digunakan di bidang-bidang sains. Teknik ini juga sudah banyak digunakan di permasalahan visi komputer, seperti *feature correspondence*, *object recognition*, dan *video analysis* (Cho & Lee, 2012). Tidak seperti teknik lainnya, *graph matching* memberikan fleksibilitas yang lebih terhadap model objek serta menjanjikan hasil dari proses *matching* dan *recognition* yang baik.

## Identifikasi Masalah

Mengacu pada latar belakang tersebut, didapatkan identifikasi masalah yang menjadi pokok penelitian ini, yaitu :

1. Bagaimana spesifikasi dataset yang dapat digunakan dalam penelitian ini.
2. Bagaimana *pre-processing* citra yang optimal sehingga didapatkan fitur-fitur yang sesuai untuk proses pengenalan objek citra.
3. Bagaimana cara mengimplementasikan *graph matching* agar memberikan hasil yang relevan pada pengenalan citra digital yang digunakan dalam penelitian ini.

## Batasan Masalah

Dari permasalahan yang telah diuraikan di atas, akan dilakukan pembatasan masalah penelitian sebagai berikut:

1. Dataset yang digunakan adalah kumpulan citra digital yang disediakan oleh peneliti baik berupa citra *artificial* maupun citra *real*.
2. Citra digital yang akan diujikan hanya berasal dari dataset peneliti.
3. Program dibangun menggunakan bahasa pemrograman Python.
4. Hasil dari penelitian ini hanya akan menunjukkan persentase kemiripan antara suatu citra dengan citra lainnya.

## Maksud dan Tujuan Penelitian

Maksud dari penelitian ini adalah untuk mencoba alternatif dalam pengenalan objek pada citra digital dengan menggunakan representasi metode pengolahan berbasis *graph*.

Tujuan yang ingin dicapai oleh penulis dari penelitian ini adalah:

1. Menentukan karakteristik citra digital yang relevan digunakan untuk pengenalan citra berbasis *graph* menggunakan metode *graph matching*.
2. Menghasilkan fitur-fitur yang optimal untuk menunjang proses *matching*.
3. Menghasilkan persentase kemiripan yang baik pada setiap citra digital yang diujikan pada penelitian ini.

## Manfaat Penelitian

Manfaat yang diharapkan penulis dari penelitian ini adalah:

1. Diharapkan hasil penelitian ini dapat digunakan untuk mengidentifikasi persamaan suatu citra digital dengan citra digital lainnya melalui deskripsi *graph*.
2. Dapat digunakan sebagai referensi untuk penelitian selanjutnya baik melalui perubahan dataset maupun pengembangan metode.

## Metodologi Penelitian

Penelitian yang akan dilakukan menggunakan metode sebagai berikut:

1. Studi literatur mengenai penelitian terkait dan teori pengenalan pola terhadap citra digital khususnya teknik *graph matching* dari berbagai sumber seperti buku elektronik, prosiding konferensi, artikel jurnal, dan skripsi sebelumnya.
2. Pembuatan (untuk citra *artificial*) dan pengumpulan dataset yang akan digunakan dalam penelitian.
3. Mengimplementasikan algoritma untuk proses-proses ekstraksi ciri dan pengenalan pola untuk menentukan kemiripan citra digital.
4. Melakukan evaluasi hasil dari uji coba pengenalan citra menggunakan teknik berbasis *graph*.

## Sistematika Penulisan

Sistematika penulisan dalam skripsi ini adalah sebagai berikut:

**BAB I PENDAHULUAN**

Bab ini menguraikan latar belakang penelitian, identifikasi masalah, batasan masalah, tujuan penelitiaa, kegunaan penelitian, metodologi penelitian, dan sistematika penulisan.

**BAB II TINJAUAN PUSTAKA**

Bab ini berisi uraian penelitian terkait yang pernah dilakukan sebelumnya mengenai pengenalan objek citra menggunakan teknik berbasis deskripsi *graph*. Pada bagian ini juga dipaparkan mengenai beberapa teori serta tulisan yang berkaitan dengan penelitian yang dilakukan.

**BAB III METODOLOGI PENELITIAN**

Pada bab ini diuraikan mengenai proses penelitian yang dilakukan. Bab ini juga menjelaskan alat serta bahan yang digunakan selama proses penelitian. Bab ini menguraikan setiap langkah yang dilakukan selama penelitian.

**BAB IV HASIL DAN PEMBAHASAN**

Bab ini membahas mengenai hasil dari pengenalan yang dilakukan menggunakan teknik *graph matching*. Pada bab ini akan dikaji tingkat akurasi dari fitur dan algoritma yang digunakan.

**BAB V SIMPULAN DAN SARAN**

Bab ini merupkan penutup dari skripsi yang berisi simpulan dan saran yang diambil dari pembahasan pada skripsi ini.

# BAB II

# TINJAUAN PUSTAKA

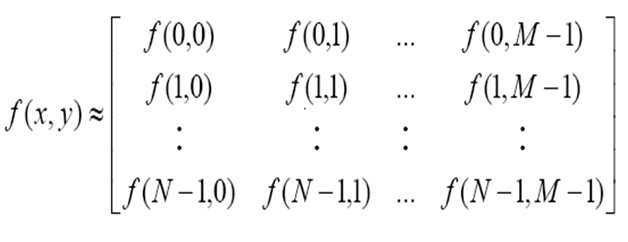
Bab ini berisi uraian mengenai penelitian terkait yang pernah dilakukan sebelumnya mengenai pengenalan objek menggunakan *graph matching*. Pada bagian ini juga dipaparkan mengenai teori serta tulisan yang berkaitan dengan penelitian yang dilakukan.

**2.1 Landasan Teori**

Berikut adalah beberapa landasan teori yang digunakan dalam proses penelitian ini.

### 2.1.1 Citra Digital

Citra digital diartikan sebagai suatu fungsi intensitas cahaya dua dimensi, yang dinyatakan oleh , di mana nilai atau amplitudo dari 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).



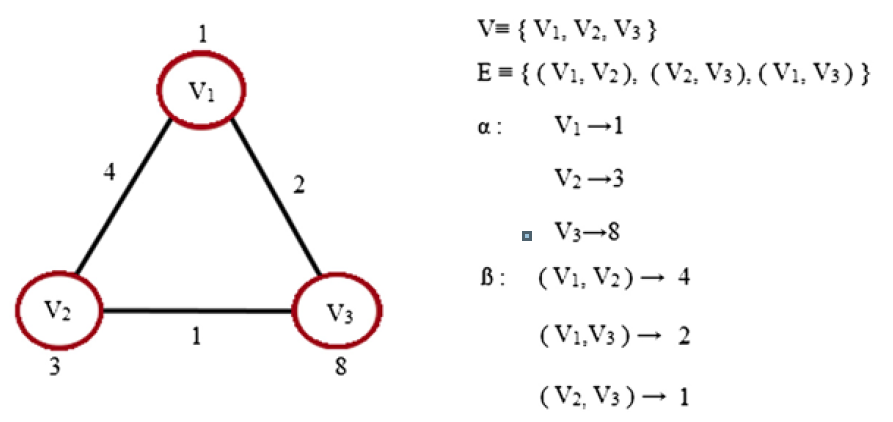
Gambar 2.1 Representasi Matriks untuk Citra Digital

### 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, sis 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).



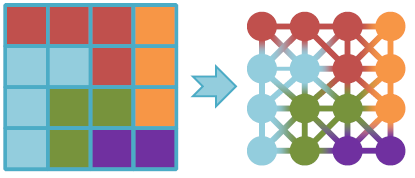
*Gambar 2. 2 Contoh Attributed Relational Graph*

### 2.1.3 Segmentasi Berbasis *Graph*

Sebelum dibangun sebuah RAG diperlukan pengelompokkan piksel-piksel yang memiliki kemiripan 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 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.

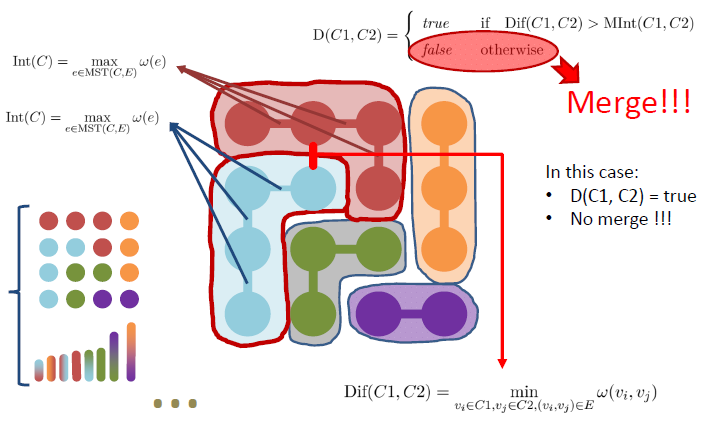
Melalui pendekatan *graph*. Dapat didefinisikan sebagai *graph* tidak berarah dengan simpul , dan sisi menunjukkan hubungan sepasang piksel yang bertetanggan. Dengan kata lain merupakan piksel-piksel dalam sebuah citra dan bobot dari sebuah sisi yang diukur berdasar intensitas perbedaan sepasang piksel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



*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 |



*Gambar 2. 4 Penggambaran algoritma (sumber : presentasi HCI-KDD)*

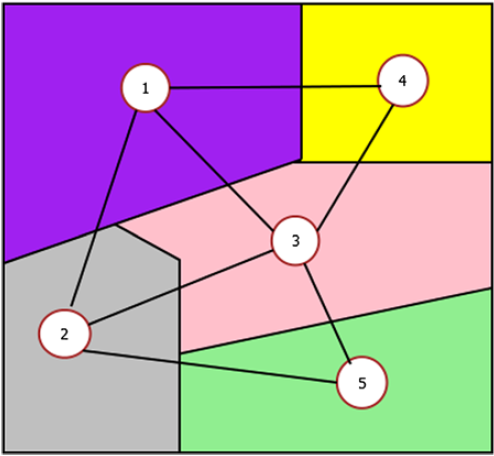
Ket :

* Nilai didapatkan dari
* , sebagai fungsi pembatas berbasis ukuran komponen. merupakan parameter konstan, semakin besar nilai menyebabkan ukuran komponen yang semakin besar. bukanlah nilai minimum komponen.

### 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).

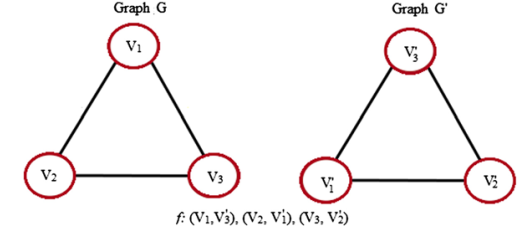
*Region Adjacency Graph* (RAG) menghubungkan bagian-bagian citra yang telah terpartisi melalui proses sebelumnya. Manfaat utama dari RAG yaitu dapat menunjukkan “*spatial view*” dari citra. Salah satu cara untuk merepresentasikan RAG yang berisi kumpulan simpul dari setiap partisi.



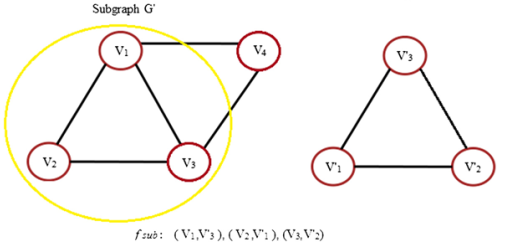
### 2.1.5 Graph Matching

*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*:

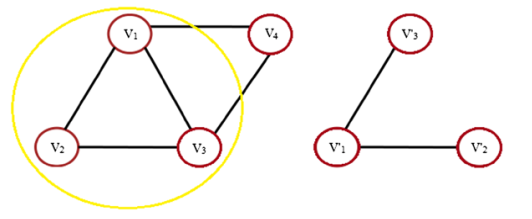
1. *Graph Isomorphism*. Proses ini menemukan struktur yang tepat antar dua buah objek *graphs*. Di dalamnya terjadi korespondensi satu-satu antar simpul dan sisi dari kedua *graph*.



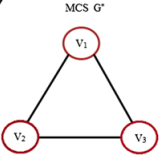
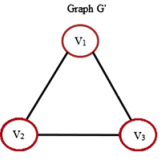
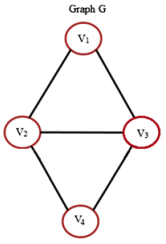
1. *Subgraph Isomorphism*. Proses ini menemukan sebuah *subgraph*  dalam sebuah *graph* . Jika simpul-simpul yang berhubungan di dalam graph dihapus, akan ditemukan sebuah *subgraph* .



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



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.*



*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*.

### 2.1.8 F1 - Score

*F1 score* (atau *F-Measure*) adalah sebuah metode pengukuran pada akurasi sebuah pengujian. Metode ini bergantung pada nilai *precision* (*p*)dan *recall* (*r*)dari hasil uji coba untuk mendapat nilai *F* tersebut. *p* merupakan jumlah dari hasil *true positive* dibagi jumlah seluruh hasil *positive* dari setiap kelas, dan *r* merupakan jumlah dari *true positive* dibagi dengan jumlah dari seluruh sample yang relevan. Secara umum metode di atas dapat dinotasikan sebagai berikut:

Sehingga nilai dari *F1 Score* dinotasikan sebagai:

**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)

*The article describes a novel method for determining similarity between histological images through graph-theoretic description and matching, for the purpose of content-based retrieval. A higher order (region-based) graph-based representation of breast biopsy images has been attained and a tree-search based inexact graph matching technique has been employed that facilitates the automatic retrieval of images structurally similar to a given image from large databases. The result showed that the proposed method is suitable for the retrieval of similar histological images, as suggested by the experimental and evaluation results obtained in the study. It is intended for the use in Content Based Image Retrieval (CBIR)-requiring applications in the areas of medical diagnostics and research, and can also be generalized for retrieval of different types of complex images*.

1. **Algoritma Pencocokan Objek Geometri Citra Berbasis *Graph* untuk Pemilihan Kembali** (Angelia, 2011)

Metode *graph* banyak digunakan dalam proses pencocokan, salah satunya adalah dengan pencocokan citra. Dalam proses *graph matching* biasanya akan ditemukan banyak kesulitan dalam algoritma yang dilakukan. Banyaknya proses dan prosedur yang harus dilakukan lamanya komputasi dalam proses *graph matching* untuk pemilihan gambar kembali (*Image retrieval*). Untuk itu, dibuatlah sebuah algoritma baru yang lebih sederhana untuk melakukan proses pencocokan citra dengan metode graph. Algoritma ini akan mengurutkan proses pencocokan dalam beberapa phase atau K phase, setiap phase akan menjelaskan solusi yang berbeda. Dengan kemungkinan bagian yang paling mendekati benar akan diutamakan dalam proses pencariannya. Hanya selisih jumlah yang kecil dalam phase yang akan digunakan untuk proses pencocokan citra (paling optimal). Dalam pengujian ini digunakan sebuah aplikasi dalam bentuk objek geometri yang akan digunakan sebagai pencocokan untuk pemilihan gambar kembali atau *content-based image retrieval*. Dari pengujian algoritma untuk pemilihan objek ini, hasil pemilihan akan menghasilkan kecocokan dengan menampilkan objek-objek geometri. Dengan tetap menampilkan *dissimilarity* objek dalam hasil proses pemilihan kembali.

1. **Interactive Image Segmentation by Matching Attributed Relational Graphs** (Noma, Graciano, Caesar Jr, Consularo, & Bloch, 2012)

*A model-based graph matching approach is proposed for interactive image segmentation. It starts from an over-segmentation of the input image, exploiting color and spatial information among regions to propagate the labels from the regions marked by the user-provided seeds to the entire image. The region merging procedure is performed by matching two graphs: the input graph, representing the entire image; and the model graph, representing only the marked regions.T he optimization is based on discrete search using deformed graphs to efficiently evaluate the spatial information. Note that by using a model-based approach, different interactive segmentation problems can be tackled: binary and multi-label segmentation of single images as well as of multiple similar images. Successful results for all these cases are presented, in addition to a comparison between our binary segmentation results and those obtained with state-of-the-art approaches. An implementation is available at* [*http://structuralsegm.sourceforge.net/*](http://structuralsegm.sourceforge.net/)*.*

# BAB III

# METODOLOGI PENELITIAN

Bab ini berisi uraian mengenai desain sistematika penelitian pengenalan objek citra digital menggunakan *graph matching*. Pada bab ini juga akan diuraikan mengenai alat dan bahan yang digunakan selama proses penelitian.

## Desain Sistematika Penelitian

Penelitian ini dilakukan untuk membangun sebuah program analisis berbasis *graph* yang dapat digunakan untuk mengenali citra digital. Seluruh proses analisis mulai dari segmentasi, ekstraksi fitur, dan pengklasifikasian akan dilakukan dengan metode berbasis *graph*. Data yang digunakan berupa citra digital sederhana yang merepresentasikan sebuah objek. Adapun untuk proses pengujian, akan dilakukan melalui perhitungan *precision* dan *recall* dari setiap data uji. Gambar 3.1 menunjukkan metode penelitian yang dilakukan.

Gambar 3. 1 Metode Penelitian

### Studi Literatur

Pada tahap ini dilakukan studi literatur terhadap penelitian-penelitian yang terkait dengan operasi pengenalan citra berbasis *graph*, mulai dari segmentasi, ekstraksi fitur, dan pengklasifikasian. Pada tahap ini dipelajari teknik-teknik yang telah menjadi *state-of-the-art* sehingga dapat menjadi rujukan dalam pembangunan metode penelitian.

Pada tahap ini juga dipelajari bagaimana penentuan spesifikasi citra digital agar memenuhi dalam penelitian ini, sehingga diharapkan dapat bekerja optimal dalam proses pengenalan citra pada metode yang dilakukan.

## Persiapan Data

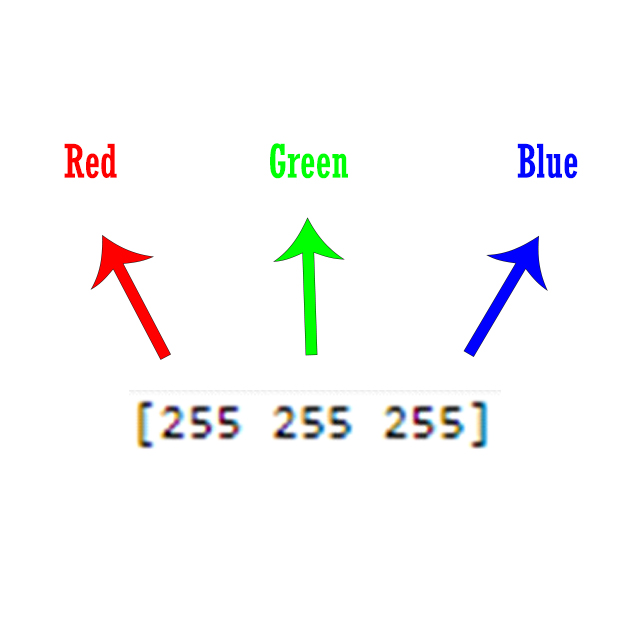
Pada tahap ini akan dilakukan persiapan data sesuai dengan batasan masalah yang telah ditentukan sebelumnya. Data sintetis yang dibuat dalam proses penelitian ini sebanyak 100 buah citra digital dengan ukuran 64 x 64 piksel. Data terdiri dari 4 kelas berbeda sehingga masing-masingnya berjumlah 25 buah. Sampel data citra ditunjukkan pada Gambar 3.2 Sampel Dataset Citra Sintetis.



Gambar 3. 2 Sampel Dataset Citra Sintetis

Data tersebut dibuat menggunakan Adobe Photoshop CS6 64 bit. Citra digambar secara manual pada halaman 64 x 64 dengan menambahkan sejumlah *shape* dengan warna yang telah ditentukan. Kemudian masing-masing citra disimpan dalam ekstensi berkas \*.jpg di dalam media penyimpanan.

Dalam bahasa pemrograman *python* penginputan citra dibantu dengan modul *io* dari *scikit-image*, kemudian setiap citra direpresentasikan dalam matriks 3 dimensi dengan nilai 64 x 64 x 3. Dimana nilai 64 pertama menunjukkan baris piksel, nilai 64 kedua kolom piksel, dan nilai 3 menunjukkan jumlah *channel* pada citra. Visualisasi dari representasi tiap piksel pada bahasa pemrograman *python* ditunjukkan oleh Gambar 3.3.



Gambar 3. 3 Representasi Satu Piksel dalam Python

### Segmentasi

Pada tahap ini akan dilakukan proses segmentasi citra menjadi bagian-bagian yang lebih sederhana. Metode yang digunakan pada penelitian ini merupakan segmentasi melalui pendekatan berbasis *graph* dengan metode yang diusulkan oleh Felzenzswalb.

Proses segmentasi ini dibantu dengan *library* dari *scikit-image* melalui sebuah fungsi *segmentation.felzenzswalb*. Kemudian dilakukan *tuning* pada parameter fungsi tersebut, untuk mendapatkan hasil segmentasi sesuai dengan persepsi jumlah *region* yang seharusnya. *Tuning* akan dilakukan kepada setiap kelas citra sehingga proses ini akan dilakukan sebanyak 4 kali (macam kelas pada citra).

*Input* pada tahap ini adalah citra digital 64 x 64 x 3. Setelah diproses, *output* dihasilkan berupa *label* dengan tipe data *np.ndarray* ukuran *64* x 64 yang menunjukkan *region* untuk setiap piksel. Selanjutnya label akan disimpan dalam format \*.csv di dalam media penyimpanan.

### Ekstraksi Fitur melalui Pembangunan RAG

Di tahap ini akan dilakukan ekstraksi fitur pada setiap citra yang telah tersegmentasi melalui pembangunan *Region Adjacency Graph* (RAG). RAG dibangun dengan menghitung nilai *centroid* dari setiap label segmentasi, menghitung nilai rata-rata RGB pada setiap *centroid*, kemudian menghubungkannya menggunakan *eucledian distance*.

*Output* dari tahap ini akan menghasilkan sebuah *graph* untuk masing-masing citra yang berperan sebagai hasil ekstraksi fitur dalam penelitian ini. Fitur-fitur yang dihasilkan dari setiap *graph* berupa:

* *Nodes*, simpul-simpul yang terbentuk
* *Total Color*, total nilai piksel RGB dari masing-masing simpul
* *Centroid*, nilai titik tengah dari masing-masing simpul
* *Mean Color*, nilai rata-rata piksel dalam RGB dari masing-masing simpul
* *Edges*, sisi-sisi yang terbentuk dari setiap simpul
* *Weight*, nilai *eucledian distance* dari masing-masing *edge*

Melalui bantuan *library* *Networkx* pada bahasa pemrograman *python*, *graph* yang dihasilkan akan memiliki tipe data berupa *class* dari *library* tersebut. Kemudian *graph* dari RAG masing-masing citra akan disimpan pada media penyimpanan dalam format berkas \*.gpickle.

### *Graph Matching*

## Alat dan Bahan Penelitian

Dalam penelitian untuk mengenali citra melalui pendekatan berbasis *graph* ini dibutuhkan perangkat keras dan perangkat lunak sebagai berikut.

1. Perangkat Keras

Perangkat keras yang digunakan adalah sebuah komputer yang memiliki spesifikasi sebagai berikut:

* *Processor* : Intel® Celeron® CPU B820 @ 1.70GHz
* RAM : 4.00 GB
* *Disk Space* : HDD 500 GB 5400 rpm

1. Perangkat Lunak

Perangkat lunak yang digunakan adalah sebagai berikut:

* Sistem Operasi : Windows 8.1 Enterprise 64-bit
* Bahasa Pemrograman : Python 2.7
* Aplikasi : Notepad++, Adobe Photoshop CS6
* Library : Numpy, Scikit-Image, Matplotlib, Networkx

# DAFTAR PUSTAKA

Akmal, Suwardi, I., & Munir, R. (2017). *Ekstraksi Graf dalam Analisis Citra Berbasis Graf.* Institut Teknologi Bandung, Sekolah Tinggi Elektro dan Informatika, Bandung.

Angelia, Y. (2011). *Algoritma Pencocokan Objek Geometri Citra Berbasis Graph untuk Pemilihan Kembali (Retrieval).* Institut Sepuluh Nopember (ITS), Jurusan Teknik Elektro-FTI, Surabaya.

Cho, M., & Lee, K. M. (2012). Progressive Graph Matching: Making a Move of Graphs via Probabilistic Voting. *Computer Vision and Pattern Recognition (CVPR).* Providence, Rhode Island: IEEE.

Cordella, L. P., Foggia, P., & Vento, M. (2001). *An Improved Algorithm for Matching Large Graphs.* Semantic Scholar.

Felzenszwalb, P. &. (2004). Efficient Graph-Based Image Segmentation. *International Journal of Computer VIsion.* doi:https://doi.org/10.1023/B:VISI.0000022288.19776.77

Munir, R. (2004). *Pengolahan Citra Digital.* Bandung: Informatika.

Noma, A., Graciano, A., Caesar Jr, R., Consularo, L., & Bloch, I. (2012). Interactive Image Segmentation by Matching Attributed Relational Graphs. *Pattern Recognition.* *45*, hal. 1159-1179. Elsevier.

Riesen, K. (2015). *Structural Pattern Recognition with Graph Edit Distance.* Olten: Springer. doi:10.1007/978-3-319-27252-8

Sharma, H., Alekseychuk, A., Leskovsky, P., Hellwich, O., Anand, R., Zerber, N., & Hufnagl, P. (2012). Determining similiarity in histological images using graph-theoretic description and matching methods for content-based image retrieval in medical diagnostics. *Diagnostic Pathology*.

Tremeau A, C. P. (2000). Regions adjacency graph applied to color image segmentation. *IEEE Trans Image Process*, *9.*