

PERBANDINGAN METODE SURF DAN SIFT DALAM SISTEM IDENTIFIKASI TANDA TANGAN

A COMPARISON OF SURF AND SIFT METHOD ON SIGNATURE IDENTIFICATION SYSTEM

Felix Pidha Hilman

Prodi S1 Teknik Telekomunikasi Fakultas Teknik Elektro,
Universitas Telkom

felixhilman@gmail.com

Abstrak

Identifikasi tanda tangan merupakan langkah yang penting untuk menghindari tindakan pemalsuan tanda tangan. Sebuah sistem identifikasi tanda tangan sangat dibutuhkan agar pemalsuan tanda tangan tidak merugikan orang lain. Metode ekstraksi ciri *Speed Up Robust Features* dan *Scale Invariant Feature Transform* akan sesuai digunakan untuk sistem tersebut. Selain itu didukung juga penggunaan *k-Nearest Neighbour* untuk proses klasifikasi.

Data yang digunakan di dalam penelitian ini adalah tanda tangan dari 10 orang dengan masing-masing memberikan 30 tanda tangan. Jumlah total 300 tanda tangan akan dibagi untuk data latih sebanyak 100 buah dan data uji sebanyak 200 buah. Hasil yang diperoleh dari pengujian adalah dengan jumlah poin=100 menggunakan ekstraksi ciri SIFT, rata-rata persentase citra benar tertinggi adalah 68% dari 200 data uji yang terbagi dalam 10 kelas. Sedangkan dengan ekstraksi ciri SURF dengan jumlah poin=125, rata-rata persentase citra benar tertinggi adalah 68% dari 200 data uji yang terbagi dalam 10 kelas.

Kata kunci : Pengenalan tanda tangan, ekstraksi ciri, SURF, SIFT, k-NN

Abstract

Identification of signatures is an important step to avoid signature forgery measures. A signature identification system is needed so that forged the signatures do not harm others. Feature extraction method *Speed Up Robust Features* and *Scale invariant Feature Transform* will be suitable for such a system. In addition it also supported the use of *k-Nearest Neighbour* to the classification process.

The data used in this study is the signature of 10 people with each providing 30 signatures. Total number of 300 signatures will be divided to as many as 100 pieces of training data and test data as much as 200 pieces. The results obtained from testing is the number of points = 100 using SIFT feature extraction, the highest average percentage of correct image is 68% of the 200 test data are divided into 10 classes. While the SURF feature extraction with the number of points = 125, the highest average percentage of correct image is 68% of the 200 test data are divided

Keywords: Signature recognition, feature extraction, SURF, SIFT, k-NN

1. Pendahuluan

Tanda tangan dapat disebut juga bukti verifikasi seseorang. Namun dengan perkembangan teknologi yang semakin pesat, maka perlu adanya sistem yang secara cepat dan tepat mengenali pola tanda tangan dengan memanfaatkan sebuah citra digital. Selain itu juga untuk mencegah adanya tindakan pemalsuan tanda tangan, yang tentunya merugikan pihak yang bersangkutan.

Metode ekstraksi ciri *Speed Up Robust Feature* (SURF) dan *Scale Invariant Feature Transform* (SIFT) dapat diimplementasikan dalam sebuah sistem pengenalan tanda tangan. Secara garis besar kedua ekstraksi ciri tersebut akan mendeteksi deskriptor pada citra latih yang akan disimpan sebagai database dan akan membandingkan dengan deskriptor yang ada pada citra uji. Kemudian akan diperoleh persentase kebenaran yang merupakan hasil dari proses klasifikasi k-NN dengan memanfaatkan perhitungan jarak antara deskriptor latih dan uji. Dari hasil tersebut akan diketahui kinerja masing-masing ekstraksi ciri yang dapat dijadikan parameter untuk membuat sistem pengenalan tanda tangan.

Untuk membantu menyelesaikan tugas akhir ini, penulis menggunakan Matlab dalam pembuatan program SURF dan SIFT. Selain itu mencari referensi tentang ekstraksi ciri SURF, SIFT dan k-NN juga dilakukan untuk menunjang tugas akhir ini.

2. Dasar Teori dan Metodologi

2.1 Pre-Processing

Pre-processing merupakan tahap awal yang dilakukan ketika menerima citra masukkan. Umumnya tahap ini akan menyeragamkan ukuran, warna dari citra masukkan, yang selanjutnya diproses pada tahap berikutnya. Selain untuk memperbaiki kualitas dari citra masukkan, tahap pre-processing ini juga bertujuan untuk memanipulasi citra masukkan sehingga sesuai yang diinginkan. Pre-processing yang digunakan dalam tugas akhir ini sebagai berikut:

2.1.1 Proses Cropping

Tahapan awal untuk pre-processing adalah dengan melakukan *cropping*. Proses *cropping* ini bertujuan agar seluruh citra memiliki ukuran yang sama sehingga akan lebih mudah dalam proses ekstraksi ciri karena semua citra, baik citra latih maupun citra uji, memiliki ukuran yang sama.

2.1.2 Proses Grayscale

Proses pengolahan citra yang pertama kali dilakukan terhadap suatu citra dalam sistem ini adalah *grayscale*. Citra RGB yang ada akan dikonversikan terlebih dahulu ke dalam citra hitam-putih (*grayscale*), hal ini dilakukan agar di dalam perhitungan untuk mendapatkan suatu fitur dari suatu citra lebih mudah.

2.2 Speed Up Robust Feature^[1]

Algoritma SURF pertama kali dipublikasikan oleh peneliti dari ETH Zurich, Herbert Bay pada tahun 2006. Dalam pengembangannya Herbert Bay juga dibantu oleh dua rekannya yaitu Tinne Tuytelaars dari Katholieke Universiteit Leuven dan Luc Van Gool. SURF mampu mendeteksi fitur lokal suatu citra dengan handal dan cepat. Algoritma ini terinspirasi dari *Scale Invariant Feature Transform* (SIFT) yang lebih dulu muncul pada 1999,

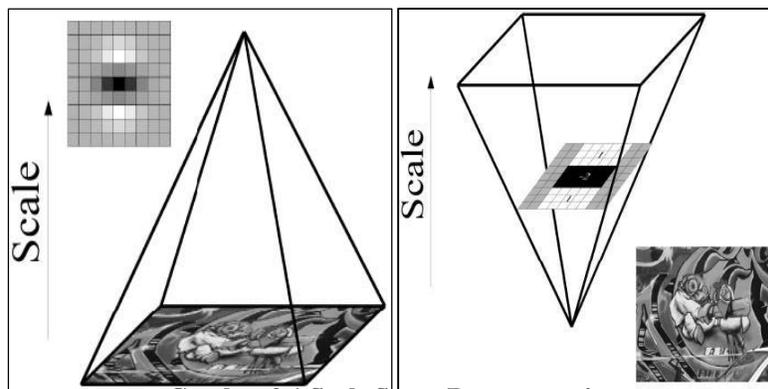
terutama pada tahap *scale space representation*. Algoritma SURF menggunakan penggabungan algoritma citra integral (*integral image*) dan *blob detection* berdasarkan determinan dari matrik Hessian.

2.2.1 Interest Point Detector

Deteksi titik perhatian (*interest point*) digunakan untuk memilih titik yang mengandung banyak informasi dan sekaligus stabil terhadap gangguan lokal atau global dalam citra digital. Dalam algoritma SURF, dipilih detektor titik perhatian yang mempunyai sifat invarian terhadap skala, yaitu *blob detection*. Blob merupakan area pada citra digital yang memiliki sifat yang konstan atau bervariasi dalam kisaran tertentu.

2.2.2 Scale Space Representation

Dengan ukuran citra yang berbeda-beda, akan sangat sulit bagi kita untuk membandingkan fitur-fitur yang terdapat pada citra tersebut. Maka dari itu, diperlukan suatu proses yang menangani perbedaan ukuran dengan menggunakan metode perbandingan skala. Dalam metode ini, kita menggunakan *scale space* (Gambar 2.1) di mana citra diimplementasikan dalam bentuk sebuah *image pyramid* (Lowe DG, 2004). Citra secara berulang akan diperhalus (*smoothing*) dengan fungsi Gaussian dan secara beruntun dengan cara *sub-sampling* untuk mencapai tingkat tertinggi pada piramida. Dengan menggunakan *integral image*, perhitungan ini tidak perlu dilakukan secara iteratif dengan menggunakan filter yang sama, tetapi dapat filter dengan ukuran sembarang ke dalam beberapa skala citra yang berbeda.



Gambar 2.1 Scale Space Representation

2.5.3 Feature Description

Fitur didefinisikan sebagai bagian yang mengandung banyak informasi suatu citra, dan fitur digunakan sebagai titik awal untuk algoritma deteksi objek. Tujuan dari proses deteksi fitur ini adalah untuk mendapatkan deskripsi dari fitur-fitur dalam citra yang diamati. Langkah pertama (Bay H et al, 2008) adalah melihat orientasi yang dominan pada titik perhatian yang terdapat dalam citra, kemudian membangun suatu area yang akan diambil nilainya dan mencari fitur korespondensi pada citra pembandingan. Dalam penentuan orientasi suatu citra kita menggunakan filter wavelet Haar, disini dapat ditentukan tingkat kemiringan suatu fitur yang diamati. Selanjutnya untuk deskripsi fitur dalam algoritma SURF.

2.5.4 Feature Matching and Recognition

Dalam tahap ini, kita membandingkan fitur hasil perhitungan proses sebelumnya (**Gambar 2.6**) tetapi hanya bila terdapat perbedaan kontras, yang dideteksi melalui tanda dari *trace* matriks Hessian. Dengan cara ini, biaya komputasi dari algoritma SURF bisa dikatakan sangat **minim**.



Gambar 2.2 Feature Matching

2.6 Scale Invariant Feature Transform^{[5][6]}

Pada tahun 1999, David G. Lowe seorang peneliti dari University of British Columbia memperkenalkan suatu metode baru dalam ekstraksi fitur dari suatu citra. Metode ekstraksi fitur ini disebut sebagai *Scale Invariant Feature Transform* (SIFT). Dengan menggunakan SIFT ini, suatu citra akan diubah menjadi vektor fitur lokal yang kemudian akan digunakan sebagai pendekatan dalam mendeteksi objek yang dimaksud.

Sebagai metode ekstraksi fitur pada pengenalan objek, SIFT ini memiliki beberapa kelebihan sebagai berikut:

- Hasil ekstraksi fitur bersifat invariant terhadap ukuran, translasi dan rotasi dua dimensi.
- Hasil ekstraksi bersifat invariant sebagian terhadap perubahan iluminasi dan perubahan sudut pandang tiga dimensi.
- Mampu mengekstrak banyak *keypoint* dari citra yang tipikal
- Hasil ekstraksi fitur benar-benar mencirikan secara khusus (*distinctive*).

Dengan kelebihan-kelebihan tersebut, penggunaan metode SIFT banyak dikembangkan untuk aplikasi pengenalan objek. Secara garis besar, algoritma yang digunakan pada metode SIFT terdiri dari empat tahap, yaitu:

- Mencari Nilai Ekstrim Pada Skala Ruang
- Mencari *Keypoint*
- Penentuan Orientasi
- Descriptor Keypoint*

Setelah melalui tahapan tersebut maka akan diperoleh fitur-fitur lokal yang digunakan sebagai *descriptor* (penciri) dari suatu objek untuk diolah lebih lanjut.

2.7 k-Nearest Neighbour^[4]

Prinsip kerjak *k-Nearest Neighbor* (k-NN) adalah mencari jarak terdekat antara data yang akan dievaluasi dengan K tetangga (*neighbor*) terdekatnya dalam data pelatihan. Teknik ini termasuk dalam kelompok klasifikasi nonparametric. Di sini kita tidak memperhatikan distribusi dari data yang ingin kita kelompokkan. Teknik ini sangat sederhana dan mudah diimplementasikan. Mirip dengan teknik klustering, kita mengelompokkan suatu data baru berdasarkan jarak data baru itu ke beberapa data/tetangga (*neighbor*) terdekat. Tujuan algoritma k-NN adalah mengklasifikasikan obyek baru berdasarkan atribut dan *training sample*. Clasifier tidak menggunakan model apapun untuk dicocokkan dan hanya berdasarkan pada memori. Diberikan titik *query*, akan ditemukan sejumlah k obyek atau (titik training) yang paling dekat dengan titik *query*. Klasifikasi menggunakan voting terbanyak diantara klasifikasi dari k obyek. Algoritma k-NN menggunakan klasifikasi ketetanggaan sebagai nilai prediksi dari *query instance* yang baru. Algoritma metode k-NN sangatlah sederhana, bekerja berdasarkan jarak terpendek dari *query instance* ke *training sample* untuk menentukan k-NN-nya. Nilai k yang terbaik untuk algoritma ini tergantung pada data. Secara umum, nilai k yang tinggi akan mengurangi efek noise pada klasifikasi, tetapi membuat batasan antara setiap klasifikasi menjadi semakin kabur. Nilai k yang bagus dapat dipilih dengan optimasi parameter, misalnya dengan menggunakan cross-validation. Kasus khusus dimana klasifikasi diprediksikan berdasarkan training data yang paling dekat (dengan kata lain, $k=1$) disebut algoritma Nearest Neighbor

Sebagai metode penghitungan jarak, k-NN memiliki kelebihan sebagai berikut:

- a. Tangguh terhadap *training* data yang memiliki banyak *noise*.
- b. Efektif apabila *training* datanya besar.

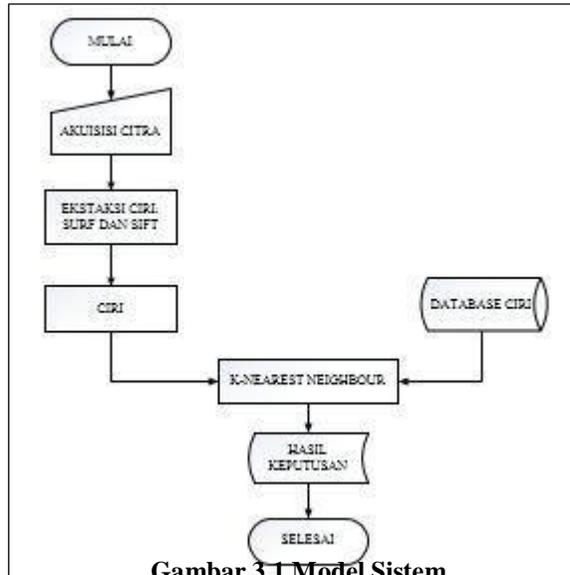
Dengan kelebihan tersebut, penggunaan algoritma k-NN akan mempermudah peneliti untuk menganalisa data-data yang berkaitan dengan jarak terdekat dalam penelitian yang akan dilakukan.

Perhitungan jarak antara data yang akan dievaluasi dengan semua pelatihan dapat dilakukan dengan ketiga perhitungan yang telah disebutkan pada awal pembahasan k-NN. Ketiga perhitungan tersebut adalah:

- a. *Euclidean Distance*
- b. *Cityblock Distance*
- c. *Cosian Distance*

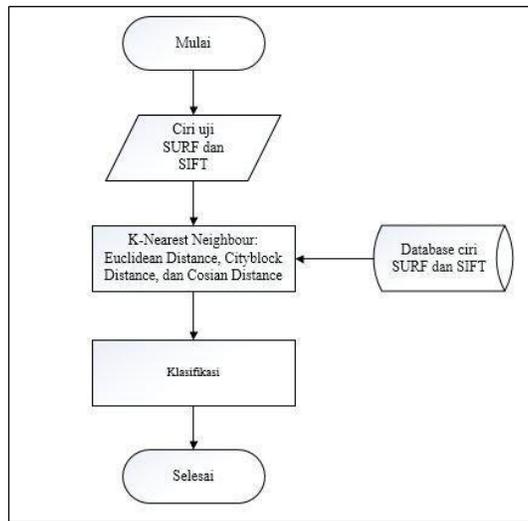
3. PERANCANGAN SISTEM

3.1 Model Sistem



Gambar 3.1 Model Sistem

3.2 Flowchart k-Nearest Neighbour



Gambar 3.2 Flowchart K-NN

4. PENGUJIAN DAN ANALISA SISTEM

Pada bab ini akan dijelaskan mengenai cara pengujian yang merupakan bagian terpenting dalam pengerjaan tugas akhir ini. Pengujian ini dilakukan untuk mengetahui apakah sistem yang dibuat telah sesuai dengan yang direncanakan. Selain itu juga untuk mengimplementasikan ekstraksi ciri SURF dan SIFT dalam sistem identifikasi

tanda tangan. Serta untuk mengetahui kelebihan dan kekurangan masing-masing ekstraksi ciri sehingga hasil yang diperoleh dapat menjadi acuan untuk penelitian selanjutnya.

4.1 Skenario 1

Pada skenario 1 pengujian tanda tangan menggunakan ekstraksi ciri SURF dan SIFT dengan menetapkan nilai deskriptor sebesar 100. Selanjutnya dalam proses klasifikasi menggunakan k-NN dengan perhitungan jarak *Euclidean Distance*, *Cityblock Distance*, dan *Cosine Distance*. Masing-masing perhitungan jarak telah ditetapkan nilai k yaitu 1, 3, 5, 7, dan 9.

4.1.1 Pengujian dengan Ekstraksi Ciri SURF

Hasil dari pengujian dengan ekstraksi ciri SURF menggunakan *Euclidean Distance*, *Cityblock Distance*, dan *Cosine Distance* adalah sebagai berikut:

Tabel 4.1 Tabel Pengujian Ekstraksi Ciri SURF menggunakan *Euclidean Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	100	135	65	67.5%
3	200	100	100	119	81	59.5%
5	200	100	100	117	83	58.5%
7	200	100	100	116	84	58%
9	200	100	100	112	88	56%
Rata-rata persentase						59.9%

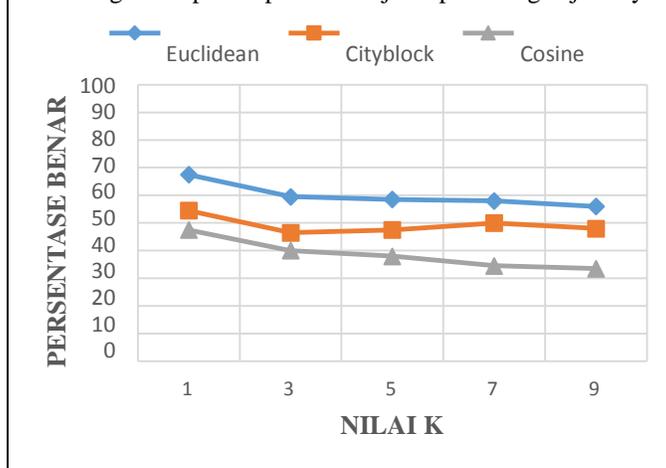
Tabel 4.2 Tabel Pengujian Ekstraksi Ciri SURF menggunakan *Cityblock Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	100	109	91	54.5%
3	200	100	100	93	107	46.5%
5	200	100	100	95	105	47.5%
7	200	100	100	100	100	50%
9	200	100	100	96	104	48%
Rata-rata persentase						49.3%

Tabel 4.3 Tabel Pengujian Ekstraksi Ciri SURF menggunakan *Cosine Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	100	95	105	47.5%
3	200	100	100	80	120	40%
5	200	100	100	76	124	38%
7	200	100	100	69	131	34.5%
9	200	100	100	67	133	33.5%
Rata-rata persentase						38.7%

Berdasarkan hasil pengujian di atas, maka dapat dibuat sebuah grafik yang menunjukkan hubungan perubahan nilai k dan persentase dengan tetap memperhatikan jenis perhitungan jarak yang digunakan.



Gambar 4.1 Grafik Hubungan Nilai K dan Persentase pada Skenario 1 menggunakan SURF

4.1.2 Pengujian dengan Ekstraksi Ciri SIFT

Hasil dari pengujian dengan ekstraksi ciri SIFT menggunakan *Euclidean Distance*, *Cityblock Distance*, dan *Cosine Distance* adalah sebagai berikut:

Tabel 4.4 Tabel Pengujian Ekstraksi Ciri SIFT menggunakan *Euclidean Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	100	122	78	61%
3	200	100	100	127	73	63.5%
5	200	100	100	128	72	64%
7	200	100	100	130	70	65%
9	200	100	100	123	77	61.5%
Rata-rata persentase						63%

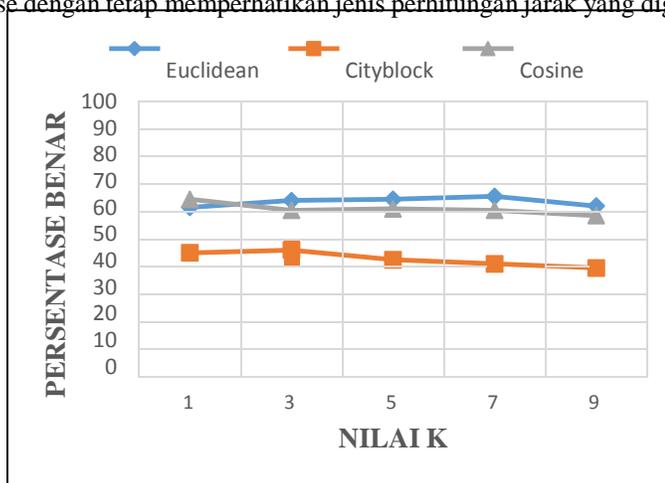
Tabel 4.5 Tabel Pengujian Ekstraksi Ciri SIFT menggunakan *Cityblock Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	100	89	111	44.5%
3	200	100	100	91	109	45.5%
5	200	100	100	84	116	42%
7	200	100	100	81	119	40.5%
9	200	100	100	78	122	39%
Rata-rata persentase						42.3%

Tabel 4.6 Tabel Pengujian Ekstraksi Ciri SURF menggunakan Cosine Distance

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	100	128	72	64%
3	200	100	100	120	80	60%
5	200	100	100	121	79	60.5%
7	200	100	100	120	80	60%
9	200	100	100	116	84	58%
Rata-rata persentase						60.5%

Berdasarkan hasil pengujian di atas, maka dapat dibuat sebuah grafik yang menunjukkan hubungan perubahan nilai k dan persentase dengan tetap memperhatikan jenis perhitungan jarak yang digunakan.

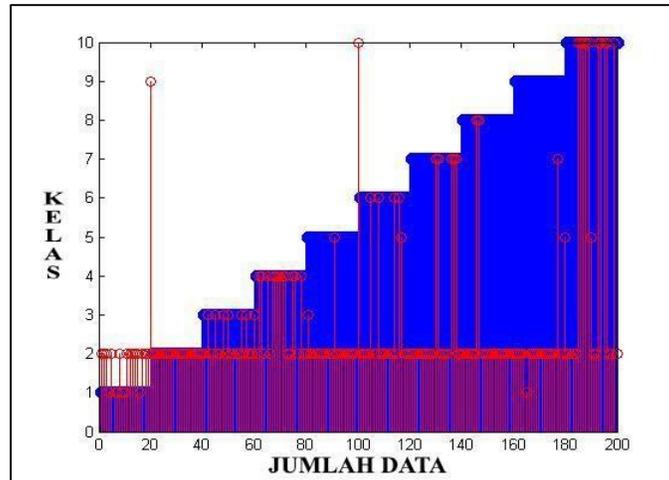


Gambar 4.2 Grafik Hubungan Nilai K dan Persentase pada Skenario 1 menggunakan SIFT

4.2 Analisa Pengujian Skenario 1

Pada pengujian skenario 1, baik menggunakan ekstraksi ciri SURF dan SIFT, seluruh hasil dapat dilihat pada tabel di atas. Ketika menggunakan ekstraksi ciri SURF, rata-rata persentase tertinggi diperoleh ketika menggunakan perhitungan jarak *Euclidean Distance*, yaitu sebesar 59.9%. Sedangkan rata-rata persentase terendah sebesar 38.7% dengan menggunakan perhitungan jarak *Cosine Distance*. Perubahan nilai k pada setiap pengujian juga berpengaruh terhadap hasil yang diperoleh. Untuk k=1 akan memiliki persentase benar lebih tinggi dibandingkan nilai k yang lainnya ketika menggunakan ekstraksi ciri SURF, berlaku untuk semua perhitungan jarak. Pada pengujian menggunakan ekstraksi ciri SIFT, rata-rata persentase tertinggi juga ketika menggunakan perhitungan jarak *Euclidean Distance* sebesar 63%. Sedangkan persentase terendah adalah ketika menggunakan *Cityblock Distance* yaitu sebesar 42.3%.

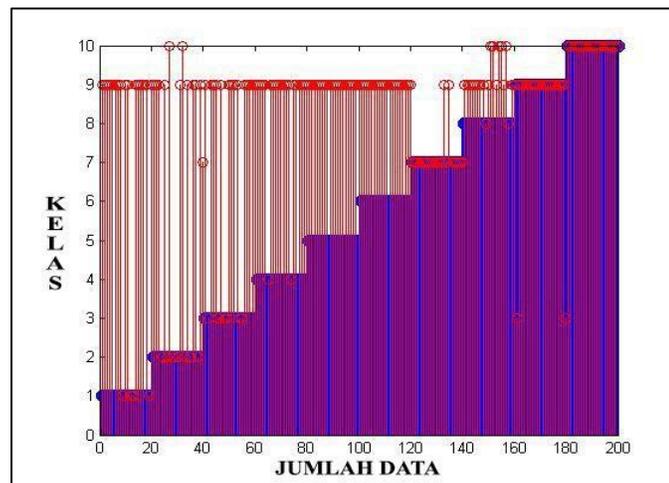
Di dalam tabel juga terdapat kolom jumlah salah yang menunjukkan bahwa suatu citra tidak sesuai dengan kelas yang telah ditentukan. Pengujian dengan ekstraksi ciri SURF menggunakan *Cosine Distance* memperoleh jumlah salah tertinggi yaitu 133 citra. Hasil tersebut dapat dilihat pada gambar di bawah ini.



Gambar 4.3 Jumlah Salah Tertinggi pada Pengujian SURF, N=100

Pada gambar di atas dapat dilihat bahwa sebagian besar citra uji tidak sesuai dengan kelompok kelas yang telah ditentukan. Kelas kedua menjadi tujuan sebagian besar citra uji. Ketidaksesuaian kelas dapat disebabkan karena pola tanda tangan pada database kelas kedua juga terdapat pada citra uji ketika dilakukan pengujian. Sehingga menyebabkan banyak citra uji yang salah dan tidak sesuai dengan kelasnya masing-masing.

Sedangkan dengan pengujian SIFT menggunakan *Cityblock Distance* jumlah salah tertinggi mencapai 122 citra. Kondisi tersebut dapat dilihat pada gambar di bawah ini.



Gambar 4.4 Jumlah Salah Tertinggi pada Pengujian SIFT, N=100

Berbeda dengan pengujian SURF, pada pengujian SIFT kelas kesembilan menjadi tujuan sebagian besar citra uji. Pola tanda tangan pada database kelas kesembilan menyebabkan ketidaksesuaian pengelompokkan kelas. Sehingga citra uji tidak masuk pada kelas yang seharusnya.

4.3 Skenario 2

Pada skenario 2 nilai deskriptor berbeda dengan skenario 1, yaitu 125. Masih dengan dua ekstraksi ciri yaitu SURF dan SIFT. Selanjutnya dalam proses klasifikasi menggunakan k-NN dengan perhitungan jarak

Euclidean Distance, *Cityblock Distance*, dan *Cosine Distance*. Masing-masing perhitungan jarak telah ditetapkan nilai k yaitu 1, 3, 5, 7, dan 9.

4.3.1 Pengujian dengan Ekstraksi Ciri SURF

Hasil dari pengujian dengan ekstraksi ciri SURF menggunakan *Euclidean Distance*, *Cityblock Distance*, dan *Cosine Distance* adalah sebagai berikut:

Tabel 4.7 Tabel Pengujian Ekstraksi Ciri SURF menggunakan *Euclidean Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	125	144	56	72%
3	200	100	125	142	58	71%
5	200	100	125	133	67	66.5%
7	200	100	125	134	66	67%
9	200	100	125	127	73	63.5%
Rata-rata persentase						68%

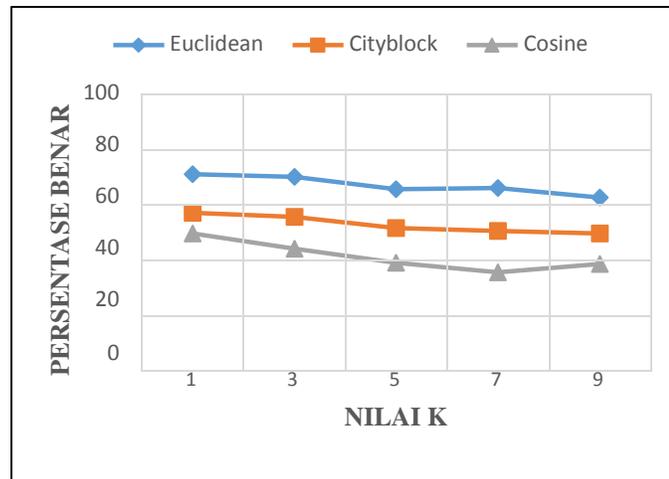
Tabel 4.8 Tabel Pengujian Ekstraksi Ciri SURF menggunakan *Cityblock Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	125	116	84	58%
3	200	100	125	113	87	56.5%
5	200	100	125	105	95	52.5%
7	200	100	125	103	97	51.5%
9	200	100	125	101	99	50.5%
Rata-rata persentase						53.8%

Tabel 4.9 Tabel Pengujian Ekstraksi Ciri SURF menggunakan *Cosine Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	125	101	99	50.5%
3	200	100	125	90	110	45%
5	200	100	125	80	120	40%
7	200	100	125	73	127	36.5%
9	200	100	125	79	121	39.5%
Rata-rata persentase						42.3%

Berdasarkan hasil pengujian di atas, maka dapat dibuat sebuah grafik yang menunjukkan hubungan perubahan nilai k dan persentase dengan tetap memperhatikan jenis perhitungan jarak yang digunakan.



Gambar 4.5 Grafik Hubungan Nilai K dan Persentase pada Skenario 2 menggunakan SURF

4.3.2 Pengujian dengan Ekstraksi Ciri SIFT

Hasil dari pengujian dengan ekstraksi ciri SIFT menggunakan *Euclidean Distance*, *Cityblock Distance*, dan *Cosine Distance* adalah sebagai berikut:

Tabel 4.10 Tabel Pengujian Ekstraksi Ciri SIFT menggunakan *Euclidean Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	125	111	89	55.5%
3	200	100	125	114	86	57%
5	200	100	125	114	86	57%
7	200	100	125	109	91	54.5%
9	200	100	125	103	97	51.5%
Rata-rata persentase						55.1%

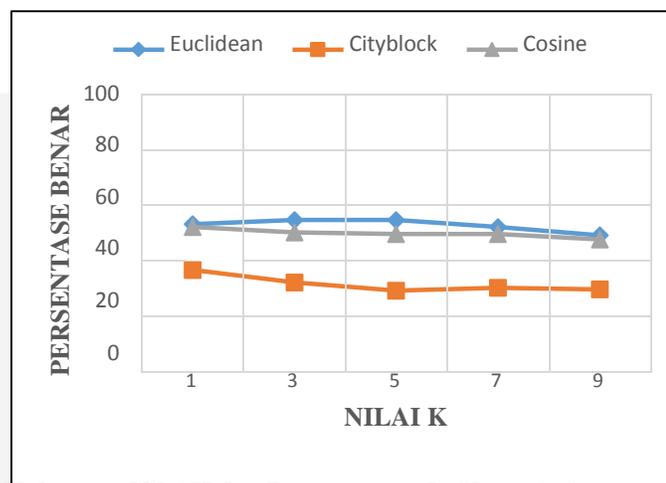
Tabel 4.11 Tabel Pengujian Ekstraksi Ciri SIFT menggunakan *Cityblock Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	125	78	122	39%
3	200	100	125	69	131	34.5%
5	200	100	125	63	137	31.5%
7	200	100	125	65	135	32.5%
9	200	100	125	64	136	32%
Rata-rata persentase						33.9%

Tabel 4.12 Tabel Pengujian Ekstraksi Ciri SURF menggunakan *Cosine Distance*

K	Jumlah Data Uji	Jumlah Data Latih	Jumlah Point	Jumlah Benar	Jumlah Salah	Persentase Benar
1	200	100	125	109	91	54.5%
3	200	100	125	105	95	52.5%
5	200	100	125	104	96	52%
7	200	100	125	104	96	52%
9	200	100	125	100	100	50%
Rata-rata persentase						52.2%

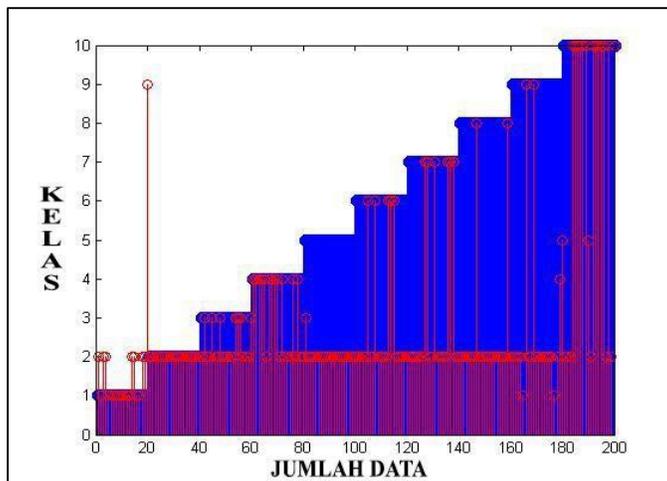
Berdasarkan hasil pengujian di atas, maka dapat dibuat sebuah grafik yang menunjukkan hubungan perubahan nilai k dan persentase dengan tetap memperhatikan jenis perhitungan jarak yang digunakan.

**Gambar 4.6 Grafik Hubungan Nilai K dan Persentase pada Skenario 2 menggunakan SIFT**

4.4 Analisa Pengujian Skenario 2

Pada pengujian skenario 2, baik menggunakan ekstraksi ciri SURF dan SIFT, seluruh hasil dapat dilihat pada tabel di atas. Ketika menggunakan ekstraksi ciri SURF, rata-rata persentase tertinggi diperoleh ketika menggunakan perhitungan jarak *Euclidean Distance*, yaitu sebesar 68%. Sedangkan rata-rata persentase terendah sebesar 42.3% dengan menggunakan perhitungan jarak *Cosine Distance*. Perubahan nilai k pada setiap pengujian juga berpengaruh terhadap hasil yang diperoleh. Untuk k=1 akan memiliki persentase benar lebih tinggi dibandingkan nilai k yang lainnya ketika menggunakan ekstraksi ciri SURF, berlaku untuk semua perhitungan jarak. Pada pengujian menggunakan ekstraksi ciri SIFT, rata-rata persentase tertinggi juga ketika menggunakan perhitungan jarak *Euclidean Distance* sebesar 55.1%. Sedangkan persentase terendah adalah ketika menggunakan *Cityblock Distance* yaitu sebesar 33.9%.

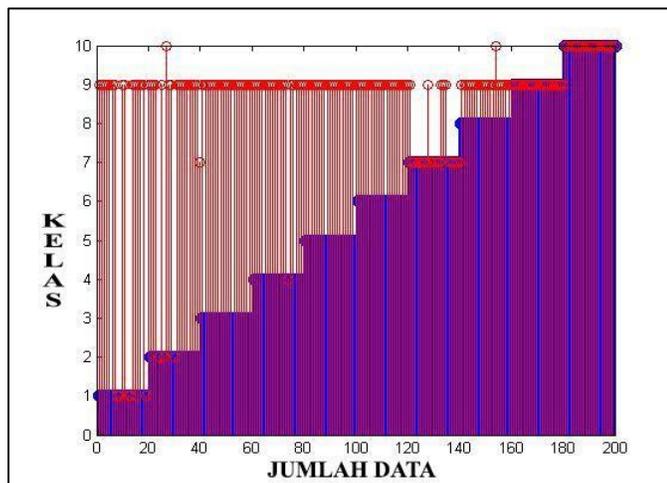
Di dalam tabel juga terdapat kolom jumlah salah yang menunjukkan bahwa suatu citra tidak sesuai dengan kelas yang telah ditentukan. Pengujian dengan ekstraksi ciri SURF menggunakan *Cosine Distance* memperoleh jumlah salah tertinggi yaitu 121 citra. Hasil tersebut dapat dilihat pada gambar di bawah ini.



Gambar 4.7 Jumlah Salah Tertinggi pada Pengujian SURF, N=125

Pada gambar di atas dapat dilihat bahwa sebagian besar citra uji tidak sesuai dengan kelompok kelas yang telah ditentukan. Kelas kedua menjadi tujuan sebagian besar citra uji. Ketidakesesuaian kelas dapat disebabkan karena pola tanda tangan pada database kelas kedua juga terdapat pada citra uji ketika dilakukan pengujian. Sehingga menyebabkan banyak citra uji yang salah dan tidak sesuai dengan kelasnya masing-masing.

Sedangkan dengan pengujian SIFT menggunakan *Cityblock Distance* jumlah salah tertinggi mencapai 122 citra. Kondisi tersebut dapat dilihat pada gambar di bawah ini.



Gambar 4.4 Jumlah Salah Tertinggi pada Pengujian SIFT, N=125

Berbeda dengan pengujian SURF, pada pengujian SIFT kelas kesembilan menjadi tujuan sebagian besar citra uji. Pola tanda tangan pada database kelas kesembilan menyebabkan ketidakesesuaian pengelompokkan kelas. Sehingga citra uji tidak masuk pada kelas yang seharusnya.

5. KESIMPULAN

Berdasarkan analisa dan evaluasi hasil penelitian, maka dapat diambil kesimpulan bahwa perubahan jumlah point sangat berpengaruh pada jumlah citra benar. Untuk ekstraksi ciri SURF, semakin tinggi jumlah point maka jumlah citra benar akan semakin tinggi. Sedangkan untuk ekstraksi ciri SIFT berlaku sebaliknya. Semakin tinggi

jumlah point, maka jumlah citra benar akan semakin rendah. Perubahan nilai k juga berpengaruh pada jumlah citra benar, hal ini berlaku untuk semua perhitungan jarak. Semakin tinggi nilai k , maka jumlah citra benar akan semakin berkurang. Dengan melihat hasil pengujian yang telah diperoleh, maka ekstraksi ciri SIFT lebih baik dibandingkan ekstraksi ciri SURF berdasarkan ketepatan dalam mengelompokkan citra.

DAFTAR PUSTAKA

- [1] Bay, Herbert., Tinne Tuytelaars, & Luc Van Gool. (2006). "*SURF: Speed Up Robust Features*". Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346-359.
- [2] Kadir, Abdul & Adhi Susanto. (2013). TEORI DAN APLIKASI PENGOLAHAN CITRA. Yogyakarta: Penerbit ANDI.
- [3] Kozma, Laszlo. (2008). "*k Nearest Neighbors algorithm (kNN)*". Helsinki University of Technology T-61.6020 Special Course in Computer and Information Science: 8-17.
- [4] Kustiyahningsih, Yeni., Devie Rosa Anamisa, & Nikmatus Syafa'ah. (2010). Jurnal: SISTEM PENDUKUNG KEPUTUSAN UNTUK MENENTUKAN JURUSAN PADA SISWA SMA MENGGUNAKAN METODE KNN DAN SMART. Bangkalan: Universitas Trunojoyo.
- [5] Lowe, D.G. (1999). "*Object Recognition from Local Scale-Invariant Features*". In International Conference on Computer Vision: 1150-1157.
- [6] Lowe, D.G. (2004). Jurnal: "*Distinctive Image Features from Scale-Invariant Keypoints*". Vancouver: Computer Science Departement University of British Columbia.