

## Pengenalan Angka Tulisan Tangan Menggunakan Scale Invariant Feature Transform (SIFT)

Shabran Fauzan Ahmad<sup>1</sup>, Febryanti Sthevanie<sup>2</sup>, Said Al Faraby<sup>3</sup>,

<sup>1,2,3</sup>Fakultas Informatika, Universitas Telkom, Bandung

<sup>1</sup>shabran@students.telkomuniversity.ac.id, <sup>2</sup>sthevanie@telkomuniversity.ac.id,

<sup>3</sup>saidalfaraby@telkomuniversity.ac.id

### Abstrak

Bidang *computer vision* khususnya OCR sudah ada sejak zaman dahulu dan sudah digunakan oleh manusia untuk mempermudah pekerjaan sehari-hari. Namun, untuk pengenalan tulisan tangan masih ada tantangan tersendiri, yaitu tingkat kesulitan untuk mengenali karena tulisan tangan setiap manusia banyak jenisnya bahkan untuk satu angka ditulis lebih dari satu kali dapat berbeda. Untuk itu, dibuatlah sistem pengenalan tulisan tangan menggunakan metode Scale Invariant Feature Transform (SIFT).

Metode SIFT merupakan salah satu metode *feature extraction* yang memiliki performa baik. Ditambah dengan *SVM classifier*, sistem pengenalan tulisan tangan berhasil dibuat dengan akurasi sebesar 95,4% pada data set MNIST sebanyak 10.000 data testing.

### Pendahuluan

Kemajuan teknologi yang pesat sudah banyak membantu manusia untuk melakukan tugas-tugas dan kegiatan sehari-hari. Hampir semua kegiatan dapat dibantu dengan mesin, salah satu contohnya adalah *pattern recognition* Tulisan tangan bisa dikenali oleh mesin namun tidak semudah mengenali cetakan hasil printer karena tulisan tangan manusia banyak jenisnya dan kadang tidak standar.

OCR (*optical character recognition*) sudah banyak menyelesaikan masalah rekognisi tulisan secara optikal. Rekognisi dilakukan secara *offline*. Tulisan sudah dibuat terlebih dahulu sebelum dilakukan pemindaian. Hampir semua jenis tulisan dapat dikenali baik tulisan tangan ataupun tulisan yang dicetak menggunakan *printer*. Meskipun demikian, akurasi dari OCR bergantung pada kualitas *input* yang digunakan.

Tugas akhir ini bertujuan untuk membuat sistem pengenalan tulisan tangan menggunakan Scale Invariant Feature Transform (SIFT) dan mengukur performansinya serta menganalisis hasil pengujian.

### Optical Character Recognition (OCR)

OCR adalah sistem yang dapat mengenali karakter dari hasil optical scan dengan bantuan komputer. OCR dapat mengenali karakter dari tulisan cetak ataupun tulisan tangan. Seiring dengan kemajuan teknologi OCR sudah banyak digunakan untuk membantu pekerjaan manusia. Cara kerja OCR meliputi 3 proses utama yaitu *optical scanning*, ekstraksi ciri, dan klasifikasi. OCR untuk tulisan cetak cenderung mudah untuk dilakukan berbeda dengan tulisan tangan. Tulisan tangan sulit untuk dikenali karena setiap orang memiliki cara penulisan yang berbeda-beda.

### Algoritma (Scale Invariant Feature Transform)SIFT

Scale Invariant Feature Transform (SIFT) merubah gambar menjadi kumpulan fitur lokal. Vektor fitur yang diperoleh dari SIFT unik dan tidak rentan terhadap perubahan atau kita bisa sebut *invariant*. Tahap awal dari SIFT adalah dengan mencari maksimum dan minimum lokal dari *Difference of Gaussian (DoG Pyramid)*. Untuk mengimplementasikan *DoG* gambar dari masukan dikonvolusikan secara berulang menggunakan *Gaussian kernel*. Operasi ini diulangi selama *down-sampling* masih bisa dilakukan. Setelah proses selesai terbentuklah *Gaussian Pyramid* yang direpresentasikan dengan fungsi :

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (2)$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (3)$$

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ = L(x, y, k\sigma) - L(x, y, \sigma)$$

Minimum dan maksimum lokal dari DoG ditentukan dengan membandingkan sebuah piksel dengan 26 tetangganya pada *scale space*. Untuk setiap kandidat *keypoints* interpolasi dilakukan untuk aproksimasi lokasi dengan akurat. Interpolasi dilakukan dengan

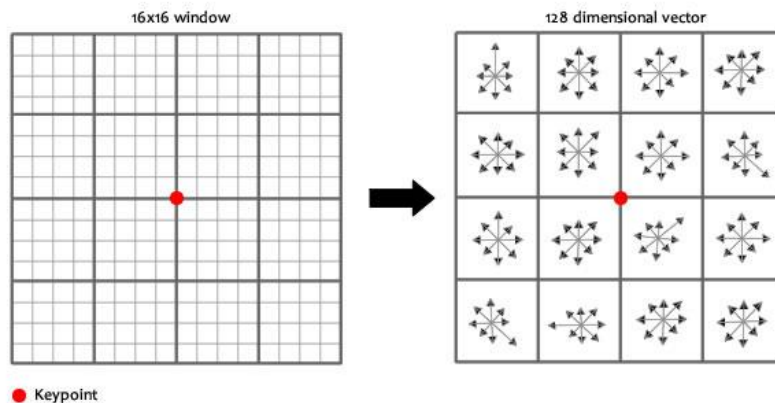
$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x$$

Proses selanjutnya adalah pemberian orientasi terhadap setiap *keypoints* berdasarkan gradient dan arahnya. Hal ini dilakukan dengan cara mengumpulkan arah gradient dan besarnya di sekitar *keypoint* lalu mencari orientasi yang terpenting pada region itu. Untuk setiap sampel gambar  $L(x,y)$ , pada *scale* saat ini dengan gradient  $m(x,y)$ , dan orientasi  $\theta(x, y)$ , dihitung menggunakan :

$$m(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^2 + (L(x, y + 1) - L(x, y - 1))^2} \tag{5}$$

$$\theta(x, y) = \tan^{-1} \left( \frac{L(x, y + 1) - L(x, y - 1)}{L(x + 1, y) - L(x - 1, y)} \right)$$

Proses terakhir adalah pembuatan *descriptor* yang hasilnya akan digunakan untuk klasifikasi. Untuk mendapatkan fitur yang baik dibuatlah *fingerprint*, yaitu representasi untuk membedakan sebuah gambar dari gambar lainnya. Untuk melakukannya dibuat sebuah *window* dengan *patch size* tertentu. Misalnya 16x16 diilustrasikan seperti gambar di bawah ini .



**Evaluasi Performansi**

Untuk mengukur performansi dilakukan pengujian terhadap 10000 data testing MNIST yang sudah menjadi standar untuk pengenalan angka tulisan tangan. Ukuran performansi ditentukan dengan cara menghitung akurasi yaitu jumlah prediksi benar dibagi dengan keseluruhan data testing. Setelah dilakukan pengujian didapatkan akurasi sebesar 95.4% dengan hasil pengenalan yang ditunjukkan pada tabel di bawah ini :

	0	1	2	3	4	5	6	7	8	9
0	<b>956</b>	0	0	5	2	1	10	3	7	13
1	1	<b>1115</b>	5	0	6	1	2	11	0	0
2	1	0	<b>996</b>	13	3	1	1	6	2	0
3	0	5	7	<b>955</b>	1	17	1	9	6	3
4	2	3	3	4	<b>924</b>	0	5	6	0	30
5	3	2	1	14	4	<b>846</b>	3	2	5	9
6	7	4	5	3	10	6	<b>932</b>	0	5	0
7	1	0	7	5	2	7	0	<b>948</b>	3	20
8	7	4	5	8	3	6	4	7	<b>939</b>	7
9	2	2	3	3	27	7	0	36	7	<b>927</b>

Dapat dilihat bahwa fitur dari SIFT umumnya dapat mengenali semua angka dengan benar kecuali beberapa angka yang mirip seperti 4 dan 9, 3 dan 5, 7 dan 9. Hal ini terjadi karena memang tulisan tangan yang bisa dikatakan salah bahkan oleh manusia sekalipun bisa salah dikenali karena memang tulisan tangan berbeda-beda.

### **Kesimpulan**

Sistem pengenalan angka tulisan tangan menggunakan SIFT dapat dibuat dengan akurasi yang tinggi yaitu 95.4%. Adapun beberapa kesalahan yang terjadi pada angka tertentu seperti 4,9,3,5,7. Untuk memperbaiki sistem ini sebaiknya dicari metode baru untuk mengenali angka tersebut yang lebih efektif atau cara lebih mudah adalah memperbaiki kualitas input citra karena pada dasarnya kualitas OCR juga tergantung pada kualitas input.



Telkom  
University

### Daftar Pustaka

- [1] R. Klette, *Concise Computer Vision*, Springer, 2014.
- [2] R. Szeliski, *Computer Vision : Algorithm and Application*, Springer, 2010.
- [3] Vladimir Vapnik, Corinna Cortes, "Support Vector Networks," *Machine Learning* 20.
- [4] L. Eikvil, *Optical Character Recognition*, 1993.
- [5] D. Lowe, "Distinctive image features from scale invariant keypoints," *International Journal of Computer Vision*, 2004.



Telkom  
University