

Deteksi Kartu Tanda Mahasiswa (KTM) Universitas Islam Negeri (UIN) Walisongo dengan *Feature Matching*

Zulaikha¹, Aries Cahya Alfarabi², Arif Rezky Aprilianto³, Maulana Misbahul Fuadi⁴,
Mohamad Tafrikan⁵, Bagus Hardiansyah⁶

¹UIN Walisongo Semarang, zulaikha@walisongo.ac.id

²UIN Walisongo Semarang, ariescahya_1908046054@student.walisongo.ac.id

³UIN Walisongo Semarang, arifr_1908046041@student.walisongo.ac.id

⁴UIN Walisongo Semarang, maulana_1908046052@student.walisongo.ac.id

⁵UIN Walisongo Semarang, tafrikan@walisongo.ac.id

⁶Universitas 17 Agustus 1945 Surabaya, bagushardiansyah@untag-sby.ac.id

DOI 10.31102/zeta.2023.8.1.1-6

ABSTRACT

The Walisongo State Islamic University student identification card (SIC) does not have much function other than just being a student identification card. In this study, SIC was used for attendance at lectures. Based on these needs, the SIC feature extraction method is used which is aimed at the SIC that is searched in the student database. The feature extraction process is based on feature detection in images with the Scale Invariant Feature Transform (SIFT). Furthermore, feature extraction validation was carried out with two images using the Root Mean Square Error (RMSE) method, which obtained an average value of 10.424.

Keywords: *feature matching, SIFT, RMSE*

ABSTRAK

Kartu tanda mahasiswa (KTM) Universitas Islam Negeri (UIN) Walisongo tidak berfungsi banyak selain hanya untuk tanda pengenalan mahasiswa. Pada penelitian ini KTM digunakan untuk proses kehadiran di perkuliahan. Berdasarkan kebutuhan tersebut, digunakan metode ekstraksi fitur KTM yang dituju dengan KTM yang dicari pada database mahasiswa. Proses ekstraksi fitur didasari dengan pendeteksian fitur pada citra dengan Scale Invariant Feature Transform (SIFT). Selanjutnya, dilakukan validasi ekstraksi fitur dengan dua citra menggunakan metode Root Mean Square Error (RMSE), yang mana diperoleh nilai rata-rata 10,424.

Kata Kunci: *ekstraksi fitur, SIFT, RMSE*

1. PENDAHULUAN

Kartu tanda mahasiswa merupakan kartu pengenalan untuk membuktikan bahwa seorang mahasiswa tersebut telah resmi menjadi mahasiswa di suatu universitas (Pamuntjar, 2015). Sistem absen di UIN masih manual ketika kuliah tatap muka, kartu tanda mahasiswa belum digunakan secara maksimal ketika kartu perpustakaan sudah dapat digunakan untuk absen serta melakukan peminjaman buku. Kurang efektifnya pembuatan kartu tanda mahasiswa yang hanya digunakan sebagai tanda pengenalan ketika di kampus, padahal jika kartu tanda mahasiswa dikembangkan bisa diperoleh fungsi yang maksimal.

Fungsi kartu mahasiswa sebaiknya dapat digunakan untuk absen ketika masuk ke kelas untuk mengikuti perkuliahan, lalu merangkap sebagai kartu perpustakaan, atau yang lebih baik lagi merangkap sebagai kartu ATM yang mana di universitas lain sudah menggunakannya.

Ketika kita mengembangkan kartu tanda mahasiswa menjadi kartu yang dapat digunakan untuk melakukan absen, maka kita dapat melakukan dengan *feature matching*. Model tersebut diimplementasikan dalam pengolahan citra untuk menentukan kesamaan citra satu dengan yang lain, dari tiap titik maupun fitur dicocokkan dengan citra yang dicari (Andono et al., 2017). Jadi ketika melakukan scan menggunakan kartu tanda mahasiswa nanti akan dicocokkan dengan kartu tanda mahasiswa yang terkoneksi dari *database*.

Berdasarkan penelitian yang berjudul “Deep Dense local Feature Matching and Vehicle Removal for Indoor Visual Localization”, lokalisasi visual yang bertujuan melokalisasi gambar kueri yang dikumpulkan sebelumnya dengan menemukan kecocokan fitur lokal untuk kueri di antara gambar di dalam ruangan parkir. Menghilangkan gambar kendaraan juga membantu untuk menangani kecocokan pada kendaraan yang identik dari tempat parkir yang berbeda. Menggunakan beberapa metode pencocokan fitur: SIFT, TimeCycle, NetVLAD, dan Ours (Park, 2022).

Hal yang baru dari artikel kami yaitu mendeskripsikan secara rinci mengenai feature matching metode Brute-Force dan FLANN dengan validasi menggunakan metode RMSE dimana dataset menggunakan data unik dari kartu tanda mahasiswa. Artikel yang kami buat bertujuan untuk mendeteksi kartu tanda

mahasiswa UIN pada gambar menggunakan feature matching untuk pengembangan kartu tanda mahasiswa agar menjadi lebih efektif.

2. TINJAUAN PUSTAKA

2.1 Penelitian Sebelumnya

Beberapa penelitian yang telah dilakukan oleh peneliti terdahulu dan berkaitan dengan metode *FEATURE MATCHING* diantaranya seperti penelitian yang dilakukan oleh Ufuk Efe dkk. (2021) yaitu artikel jurnal yang berjudul *DFM: A Performance Baseline for Deep Feature Matching* menghasilkan nilai 0,57 dan 0,80 skor keseluruhan dengan menggunakan validasi metode MMA (*Mean Matching Accuracy*) untuk ambang batas 1 dan 2 piksel pada dataset *Hpatches* yang menunjukkan hasil yang lebih baik. Kemudian penelitian dari Kyung Ho Park (2022) dengan judul *Deep Dense Local Feature Matching and Vehicle Removal for Indoor Visual Localization* yaitu mencocokkan beberapa bagian pada lahan parkir pada dataset benchmark dimana berisi 267 gambar dan dikumpulkan sebelumnya dalam 99 gambar kueri dan diambil dari 34 bagian tempat parkir menghasilkan akurasi sebesar 86,9% dan mengungguli bagian alternatif.

2.2 Feature

Feature atau fitur merupakan karakteristik tertentu pada suatu gambar yang memiliki sifat kekhasan atau keunikan tersendiri. Fitur ini digunakan sebagai titik awal dalam algoritma deteksi objek (Khoharja et al., 2017).

2.3 Corner Detection

Corner atau sudut adalah perpotongan dari dua buah edge atau titik yang menghasilkan dua arah edge dominan yang berbeda (Burger, 2008). Corner detection atau deteksi sudut adalah pendekatan yang biasa digunakan untuk mengekstrak berbagai jenis feature, mengenal karakteristik dari suatu citra, kemudian menyimpulkan isi dari sebuah citra (Bhatia & Chhabra, 2011).

2.4 Feature Detection

Feature detection atau deteksi fitur merupakan metode untuk memperoleh informasi yang bersifat abstrak atau fitur pada suatu citra, dimana fitur yang dikenali memiliki peranan penting untuk pencocokan dengan citra lainnya (Gunarso et al., 2016).

2.5 Feature Description

Feature description atau deskripsi fitur bertujuan memberikan setiap keypoint yang

ditemukan deskripsi atas area di sekitarnya. Untuk mengatasi masalah dalam pengambilan gambar, descriptor harus mempunyai ketahanan terhadap perubahan skala dan rotasi. Hal tersebut dapat dilakukan dengan algoritma KAZE, ORB, SURF, dan SIFT (Khoharja et al., 2017).

2.4.1 ORB

Oriented FAST and Rotated BRIEF atau ORB menggunakan metode FAST-9 dalam mendeteksi fitur, yang membandingkan kandidat keypoint dengan pixel lingkaran di sekelilingnya dalam radius 9 piksel. Massa dari setiap keypoint akan dihitung dengan (Khoharja et al., 2017):

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y)$$

Oleh karena itu, orientasi keypoint dapat dicari dengan $\theta = \text{atan2}(m_{01}, m_{10})$. Descriptor diperoleh dengan cara ambil 31x31 *patch* yang ada di sekitar keypoint. Buat masing-masing *patch* menjadi 5x5 *subpatch*. ORB menggunakan 2 *subpatch* untuk dinilai intensitas yang lebih besar antara *subpatch* pertama dan kedua sehingga didapat nilai biner 0 dan 1. Proses *learning* dilakukan oleh ORB untuk memperoleh 256 string biner. Pertama urutkan *mean* pasangan *subpatch* dari yang paling dekat dengan 0,5 (menunjukkan variasi yang tinggi). Kemudian, lakukan *greedy search*:

- 1) Masukkan pasangan pertama ke vektor hasil;
- 2) Ambil pasangan berikutnya dan bandingkan dengan semua pasangan di vektor R, jika mempunyai korelasi yang tinggi maka akan dibuang;
- 3) Lakukan langkah sebelum ini sampai vektor R berisi 256 pasangan.

2.4.2 SIFT

Scale Invariant Feature Transform (SIFT) diperkenalkan oleh David G. Lowe pada tahun 1999. Dalam metode ini, sebuah citra diubah menjadi vektor fitur lokal, lalu digunakan sebagai pendekatan dalam mendeteksi suatu objek. Secara umum, algoritma pada metode SIFT ini mempunyai 4 tahap (Low, 2004), antara lain:

- 1) Deteksi ruang skala extrema: citra diperkecil sebanyak oktaf σ , kemudian aplikasikan Gaussian Blur di setiap oktaf dengan level k. Difference of Gaussian (DoG) diperoleh dari perbedaan citra setiap oktaf.
- 2) Lokalisasi Keypoint: cari nilai maksimum dan minimum dari masing-masing keypoint

dengan cara membandingkan dengan 3x3x3 tetangga di level yang sama, level di atas, dan juga level di bawahnya.

- 3) Penentuan orientasi kanonik: lakukan filter pada keypoint dengan kontras rendah dan edge dengan menyusun Hessian matriks dari area keypoint dan kedua nilai eigen valuenya diambil dengan rumus:

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r}$$

Dimana trace diperoleh dari $Tr(H) = D_{xx} + D_{yy}$ dan determinan diperoleh dari $Det(H) = D_{xx}D_{yy} - (D_{xy})^2$. Jika lebih kecil dari threshold 10, maka akan disimpan dan dihitung gradien beserta orientasinya.

- 4) Deskripsi keypoint: 16x16 pixel dengan titik tengah keypoint dibagi menjadi 4x4 untuk dibuat histogram. Histogram setiap gradien memiliki ukuran 8-bin sehingga didapat descriptor 128 dimensi (Do et al., 2012).

2.6 Feature Matching

Feature matching merupakan sebuah metode yang melibatkan pendeteksian feature yang memiliki descriptor yang sama atau mirip (Khoharja et al., 2017).

2.7 Brute-Force

2.7.1 Algoritma Brute-Force

Algoritma brute-force merupakan metode yang bergerak dari kanan hingga ke kiri pada pencocokan pattern. Algoritma ini mempunyai kesederhanaan dalam menyelesaikan permasalahan tanpa suatu pemikiran yang lama. Algoritma brute-force dianggap sebagai metode terbaik dalam mencari pola karena waktu pencarian yang lebih cepat (Rahmawati et al., 2021).

2.7.2 Brute-Force Matching

Untuk proses pencarian algoritma brute force lakukan pencocokan string dengan langkah-langkah sebagai berikut (Mesran, 2014):

- 1) Algoritma brute force mulai mencocokkan pattern di awal teks
- 2) Dari kiri ke kanan, akan dilakukan pencocokkan setiap karakter pattern dengan karakter pada teks yang bersesuaian, hingga salah satu kondisi berikut terpenuhi:
 - Perbandingan karakter di pattern dan di teks tidak cocok.
 - Setiap karakter pada pattern cocok. Selanjutnya algoritma akan memberitahukan penemuan pada posisi ini.

3) Algoritma melakukan penggeseran pattern sebesar satu ke kanan dan mengulangi langkah ke 2 hingga pattern berada di ujung teks.

2.8 FLANN

Metode Fast Library Approximated Nearest Neighbor (FLANN) merupakan suatu library atau kumpulan algoritma yang bekerja untuk melakukan pencarian secara cepat, perkiraan pixel tetangga, yang terdapat di *space* dimensi yang tinggi (Hamizan & Sumiharto, 2017).

Metode Fast Library Approximated Nearest Neighbor (FLANN) digunakan untuk matching fitur SURF citra wajah data awal dengan fitur SURF citra wajah uji. Fitur SURF ini terdiri dari 3 komponen utama, yaitu: keypoint, descriptor dan vektor. Untuk satu gambar wajah citra uji terdapat cluster untuk fitur SURF. Cluster ini akan otomatis dengan menggunakan K-Nearest Neighbor (KNN) dengan tipe indeks kd tree, dimana KNN ini akan mencari jarak yang paling kecil antara vektor sampel dengan vektor pada cluster. Proses matching fitur citra wajah uji dan fitur citra wajah sampel, vektor keypoint dan descriptor pada citra wajah akan di cocokkan nilainya dengan menggunakan KNN search. KNN search mencari cluster pada citra wajah uji yang nilai vektor deskriptornya paling dekat jaraknya dengan vektor deskriptor (Oksaputri et al., 2018).

3. METODE PENELITIAN

3.1 Bahan dan Metode

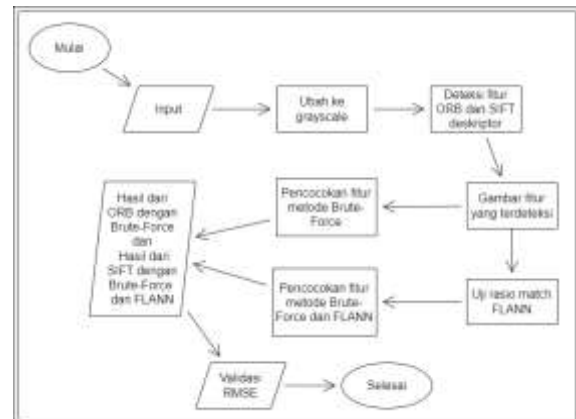
Dalam penelitian ini, peneliti menggunakan windows 10 dan 11 serta software pendukung komputasi berupa Matlab R2013a dan Google Colab, jaringan wifi dan koneksi internet.

3.2 Metode

Pada bagian ini dipaparkan tentang metode yang digunakan dalam penelitian dan juga disertai dengan pustaka yang menjadi dasar teori dalam penelitian ini, yaitu berupa penelitian sebelumnya, pengertian *feature matching*, validasi dengan menggunakan *Brute-Force* dan sebagainya. Beberapa langkah yang dilakukan oleh peneliti dalam melakukan penelitian ini yaitu studi literatur, pengumpulan dataset, pengolahan data, penerapan metode yang digunakan, simulasi, validasi hasil simulasi dan yang terakhir kesimpulan.

3.3 Flowchart

Langkah-langkah yang dilakukan dalam penelitian ini adalah sesuai dengan *flowchart* sebagai berikut

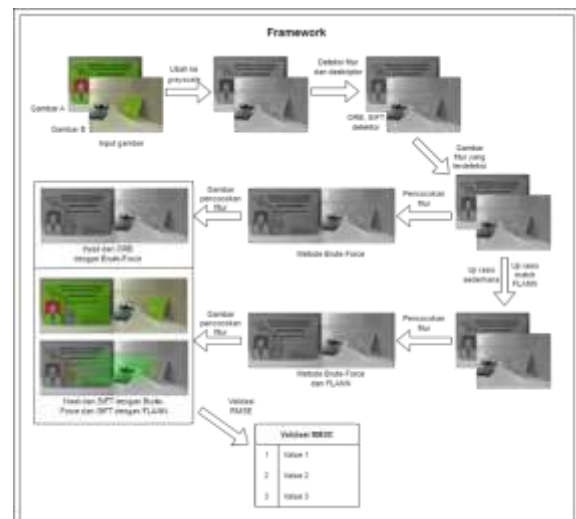


Gambar 1. Flowchart feature matching

4. HASIL PENELITIAN

4.1. Feature Matching

Berikut rangkaian alur dari proses memasukkan gambar sampai dengan proses mendapatkan hasil beserta validasinya:



Gambar 2. Framework feature matching

Feature matching menggunakan metode Brute-force serta FLANN dengan deskriptor berupa ORB dan SIFT. Hasil eksperimen menggunakan kartu tanda mahasiswa UIN sejumlah 20 mahasiswa, dengan masing-masing dibutuhkan 2 citra, sehingga total data yang diolah sejumlah 40.

4.2. Hasil Feature Matching

Pertama, mencari fitur dan deskriptor menggunakan metode ORB lalu melakukan pencocokan fitur menggunakan metode Brute-Force. Kedua, mencari fitur dan deskriptor menggunakan metode SIFT lalu melakukan uji coba pencocokan, dan pencocokan fitur menggunakan metode Brute-Force dan FLANN.

Hasil dari deteksi fitur menggunakan SIFT:



Gambar 3. Hasil deteksi fitur dengan SIFT pada KTM yang dituju



Gambar 4. Hasil deteksi fitur dengan SIFT pada KTM yang dicari

Hasil dari pencocokan fitur diperoleh seperti pada gambar berikut:



Gambar 5. Hasil dari feature matching menggunakan Brute-Force dengan ORB



Gambar 6. Hasil dari feature matching menggunakan Brute-Force dengan SIFT



Gambar 7. Hasil dari feature matching menggunakan FLANN dengan SIFT

Deteksi fitur pada metode ORB mendapatkan hasil yang lebih sedikit dari metode SIFT yang mengakibatkan hasil pencocokan fitur juga berbanding lurus dengan jumlah fitur yang dideteksi. Berdasarkan hasil dari ketiga gambar di atas menunjukkan bahwa pencocokan fitur dengan metode FLANN serta SIFT deskriptor lebih banyak memperoleh fitur yang cocok daripada metode Brute-Force. Metode FLANN maupun Brute-Force dapat melakukan pencocokan fitur pada KTM dengan baik. Sehingga kedua metode tersebut layak untuk melakukan feature matching pada dataset KTM UIN. Selanjutnya akan dihitung proses validasi menggunakan metode RMSE.

4.3 Metode RMSE (Root Mean Square Error)

Nilai RMSE dihitung menggunakan persamaan berikut:

$$RMSE = \sqrt{\frac{1}{M \times N} \sum_{x=1}^N \sum_{y=1}^M (f_1(x, y) - f_2(x, y))^2}$$

Dimana:

$f_1(x, y)$ = intensitas citra sebelum diproses

$f_2(x, y)$ = intensitas citra setelah diproses

M = panjang baris citra

N = panjang kolom citra

Hasil validasi sebagian data menggunakan Metode RMSE (Root Mean Square Error) diperoleh sebagai berikut sesuai pada tabel 1:

Tabel 1. Hasil Validasi RMSE

No	Kartu tanda mahasiswa	Hasil Validasi
1	Aries	9.2412
2	Arif	4.9180
3	Faiz	8.2580
4	Fikri	10.764
5	Firda	12.800
6	Maulana	13.164
7	Putri	11.262
8	Rizky	12.170
9	Roji	11.280
10	Winda	10.379

Berdasarkan hasil validasi diperoleh nilai RMSE paling rendah 4.9180, nilai tertinggi 13.164, dan rata-rata nilai RMSE sebesar 10.424.

5. KESIMPULAN

Deteksi KTM UIN menggunakan feature matching dapat berjalan dengan baik, dari pendeteksian fitur masing-masing KTM dengan metode ORB dan SIFT. Deskriptor SIFT memperoleh deteksi fitur lebih banyak dari hasil deteksi fitur ORB. Sedangkan pada hasil feature matching metode FLANN matcher lebih baik dari Brute-Force dalam melakukan pencocokan fitur. Proses validasi menggunakan metode Root Mean Square Error mendapatkan nilai rata-rata sebesar 10,424.

Metode feature matching dapat digunakan untuk melakukan deteksi KTM UIN pada gambar dengan lancar akan tetapi tidak semua fitur terdeteksi secara menyeluruh. Sebaiknya dilakukan lebih banyak uji coba dengan berbagai metode berbeda untuk mendapatkan metode yang benar-benar sesuai untuk kedepannya.

DAFTAR PUSTAKA

- Do, T. T., Kijak, E., Amsaleg, L., & Furon, T. (2012). Enlarging hacker's toolbox: Deluding image recognition by attacking keypoint orientations. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*.
- Gunarso, M., Muharom, L. A. Oktavianto, H. (2016). Deteksi Objek pada Gambar Menggunakan Algoritma Speeded-Up Robust Featur (SURF). *Journal of Undergraduate Thesis, Universitas Muhammadiyah Jember*.
- Hamizan, Z., & Sumiharto, R. (2017). Sistem

Pentautan Citra Udara Menggunakan Algoritme SURF dan Metode Reduksi Data. *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, 7(2), 127.

Khoharja, G., Eng, Liliana, Purbowo, A. N (2017). *Aplikasi Deteksi Nilai Uang pada Mata Uang Indonesia dengan Metode Feature Matching*. 2–6.

Kyung Ho Park (2022) *Deep Dense Local Feature Matching and Vehicle Removal for Indoor Visual Localization*, Computer Science.

Low, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 91–110.

Mesran, M. (2014). Implementasi Algoritma Brute Force Dalam Pencarian Data Katalog Buku Perpustakaan. *Majalah Ilmiah INTI*, 3(1), 100–104.

Oksaputri, R., Ernawati, E., & Andreswari, D. (2018). Implementasi Content Based Image Retrieval (CBIR) Pada Citra Batik Besurek Yang Tidak Utuh Menggunakan Metode Speeded Up Robust Features (SURF) dan Fast Library Approximated Nearest Neighbor (FLANN). *Pseudocode*, 5(2), 18–28.

Pamuntjar, Heris (2015). *Desain Sistem Cetak Kartu Tanda Mahasiswa Universitas Merdeka Malang*. GARUDA: Jurnal Teknologi dan Manajemen Informatika.

Patel, A., R. Kasat, D., Jain, S., & M. Thakare, V. (2014). Performance Analysis of Various Feature Detector and Descriptor for Real-Time Video based Face Tracking. *International Journal of Computer Applications*, 93(1), 37–41.

Pulung Nurtantio Andono, T. Sutojo, Muljono (2017) *Pengolahan Citra Digital*, Penerbit ANDI, Yogyakarta.

Rahmawati, Y., Adi Pribadi, I., & Heningtyas, Y. (2021). Penerapan Algoritma Brute Force Pada Menu Search Website “Calonku” Dalam Rangka Pemilu Berbasis Web. *Jurnal Pepadun*, 2(1), 60–70.

Ufuk Efe, Kultamis Gokalp Ince, A. Aydin Alatan (2021) *DFM: A Performace Baseline for Deep Feature Matching*, Computer Science.