

EKSTRAKSI INFORMASI MEME-INTERNET BERBAHASA INDONESIA DENGAN MESIN PENCARIAN

Fery Satria Kristianto, Endang Setyati dan Reddy Alexandro Harianto

Institut Sains dan Teknologi Terpadu Surabaya

Email: fsatria26@gmail.com, endang@stts.edu, reddy@stts.edu

Abstract

Meme images on the internet have information that can be extracted with the aim of obtaining new information that is needed by society. Information extraction is one of the sciences in recognizing unstructured text data into structured text data. In this research, we will conduct trials in recognizing the background image and text contained in the image. The position of the text that you want to recognize is at the top and bottom of the meme image. The SIFT algorithm is one of the most widely used feature extraction methods in recognizing local features of an image. The use of the SIFT algorithm in recognizing local features of the image is expected to provide good information. Meanwhile, the Jaro-Winkler distance algorithm is used for textual similarity. Trials in the pre-process in forming a database of image and word features using 1000 image data from the crawling process can be carried out. The preprocessing especially for the text extraction stage contained in the image can run well. In the information retrieval test phase, 50 images were used. The results of the information retrieval trial by utilizing the database feature formed in the study have not been maximized because the output of the system in providing results is still far from the initial hypothesis. The percentage of the similarity in the image results from the output test is query still below 50%, especially in the section on the compatibility between the image text and the background image.

Keyword : extraction, information, SIFT, jaro-winkler, internet-memes

Abstrak

Citra meme yang ada di internet memiliki informasi yang dapat digali dengan tujuan untuk mendapatkan informasi baru yang diperlukan masyarakat. Ekstraksi Informasi merupakan salah satu ilmu dalam mengenali data teks tidak terstruktur menjadi data teks terstruktur. Dalam penelitian akan melakukan uji coba dalam mengenali citra latar dan teks yang terkandung di dalam citra. Posisi teks yang hendak dikenali berada pada bagian atas dan bawah citra meme. Algoritma SIFT adalah salah satu metode ekstraksi fitur yang paling banyak digunakan dalam mengenali lokal fitur dari sebuah citra. Penggunaan algoritma SIFT dalam mengenali lokal fitur dari citra diharapkan dapat memberikan informasi yang baik. Sedangkan untuk kemiripan secara tekstual digunakan algoritma Jaro-Winkler distance. Uji coba dalam pra-proses dalam pembentukan *database* fitur citra dan kata menggunakan 1000 data citra hasil proses crawling dapat dilaksanakan. Praproses terutama untuk tahapan ekstraksi teks yang terdapat pada citra dapat berjalan baik. Pada tahap uji coba temu kembali informasi data citra yang

digunakan sebanyak 50 citra. Hasil uji coba temu kembali informasi dengan memanfaatkan *database* fitur yang terbentuk dalam penelitian, belum berjalan maksimal karena luaran dari sistem dalam memberikan hasil masih jauh dari hipotesa awal. Prosentase kemiripan citra hasil ujicoba luaran *query* masih di bawah 50%, terutama di bagian kesesuaian antara teks citra dan citra latar.

Kata kunci: ekstraksi, informasi, SIFT, jaro-winkler, meme-internet

Pendahuluan

Penyebaran meme menjadi sebuah budaya baru di masyarakat untuk menyampaikan ekspresi dari suatu ide, perilaku ataupun gaya dari satu orang ke orang lain terutama di media sosial. Istilah bahasa Inggris meme dicetuskan oleh Richard Dawkins tahun 1976 melalui bukunya yang berjudul “The Selfish Gene”(Dawkins, 2016). Meme-internet sebagai salah satu hasil produk dari teknologi digital khususnya citra gambar yang dapat memberi pengaruh positif maupun negatif.

Pengaruh ini baik dari segi gambar yang di jadikan latar belakangnya maupun kalimat teks yang menjadi bagian dari gambar tersebut. Dengan semakin banyaknya minat para kreator meme dan pengguna internet di Indonesia. Hal ini mendorong anak-anak bangsa membuat sebuah wadah di internet yang dapat banyak menampung kreatifitas meme yang terutama yang memiliki unsur komedi. Salah satu laman yang populer adalah 1 Cak.com, sebuah laman yang di dalamnya menampung berbagai macam meme yang terinspirasi dari kehidupan sehari-hari atau yang menjadi *trending topic* di media sosial.

Frase Kalimat pada citra meme dapat menjadi sebuah informasi yang dapat membantu dalam mencari tingkat kepopuleran dari citra meme tersebut di lihat dari segi banyaknya meme dengan citra latar yang mirip tersebar di internet. Dalam penelitian ini, akan dilakukan pemisahan teks dari citra latar meme agar di dapatkan data teks yang dapat di olah menjadi informasi. Sehingga muncul tantangan dalam menggali maupun melakukan ekstraksi informasi dari obyek citra. Untuk menyelesaikan kendala tersebut dilakukan teknik *Information Extraction* (IE), dimana teknik ini untuk mengambil teks pada citra meme menjadi sebuah informasi yang mudah digunakan kembali. Definisi dari ekstraksi informasi adalah suatu mekanisme dalam melakukan ekstraksi sekumpulan teks untuk mendapatkan suatu fakta-fakta. Fakta-Fakta tersebut dapat dalam bentuk events, entitas dan relationship maupun dalam bentuk informasi terstruktur (Saggion & Poibeau, 2013).

Ekstraksi informasi merupakan salah satu bagian dari Pemrosesan Bahasa Alami (Natural Language Processing). Pada penelitian ini proses ekstraksi akan dilakukan dengan menggunakan metode Optical Character Recognition (OCR), dimana proses ini dilakukan untuk mengubah pada bagian teks dari sebuah citra optis yang memiliki dokumen atau teks didalamnya(Amalia, Sharif, Haisar, Gunawan, & Nasution, 2018). Menurut Cherriet, OCR adalah sebuah aplikasi komputer yang digunakan untuk mengidentifikasi citra huruf maupun angka untuk dikonversi ke dalam bentuk yang dapat diubah.(Cheriet,M. & Kharma,N., Liu,C., Suen, 2006)

Penelitian yang di usulkan ini adalah untuk melihat kemiripan baik secara frase kata yang terkandung di dalam citra tersebut maupun obyek dari citra latarnya. Untuk ekstraksi fitur dalam mencari kemiripan citra latar, peneliti akan menggunakan salah satu algoritma pengenalan obyek yaitu SIFT (Nakashima & Kuroki, 2017). Algoritma SIFT digunakan untuk menemukan titik-titik kunci pada image dan SIFT deskriptor.

Titik kunci antar citra kemudian akan dihitung menggunakan *cosine distance*. Sedangkan kemiripan pada kata, peneliti menggunakan algoritma jaro-winkler *distance*. Metode yang di usulkan dalam penelitian ini adalah menggabungkan kedua algoritma dalam menemukan kembali informasi dari citra meme.

Metode Penelitian

Tahapan awal dalam penelitian ini adalah pembuatan dataset dengan melakukan proses *crawling* citra meme di internet. Proses ini mengumpulkan sebanyak 1000 citra meme dengan teks dalam bahasa Indonesia yang akan digunakan sebagai dataset awal. Salah satu hasil citra meme yang diperoleh seperti pada gambar 1.



Gambar 1 Contoh Citra Meme

Pada tahap selanjutnya akan dilakukan beberapa tahapan pra-proses terhadap citra hingga proses ekstraksi teks dan vector fitur pada citra diperoleh. Tahapan-tahapan pra-proses sebagai berikut:

1. Pra-proses citra

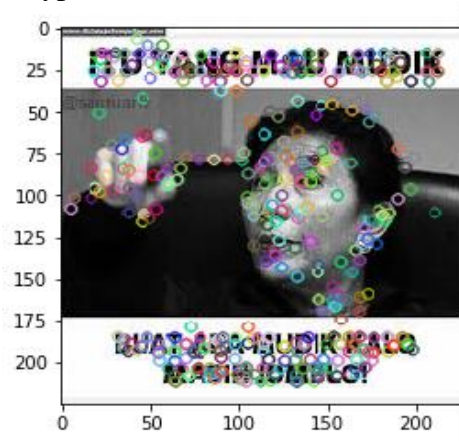
Metode pra-proses yang dilakukan adalah dengan melakukan konversi citra RGB ke *grayscale*. Hal ini dilakukan untuk mempercepat proses pada tahapan selanjutnya karena dengan dengan nilai hanya derajat keabuan, proses ekstraksi akan lebih cepat dilakukan. Hal ini untuk merubah intensitas informasi warna dari RGB (16 bit) menjadi 8 bit. Sedangkan untuk segmentasi akan menggunakan metode *Thresholding* untuk mendapatkan wilayah yang mengandung teks dan non teks. Pada tahapan ini dilakukan proses morfologi untuk dapat mendeteksi wilayah teks pada citra yang memiliki beragam warna latar sehingga diperlukan proses morfologi gradien dan dilasi pada citra biner dan kemudian dilakukan proses *inverted* pada citra. Sehingga wilayah teks yang dikenali dalam warna putih dan background berwarna hitam dapat dilakukan transposisi sehingga nantinya wilayah teks berwarna hitam. Proses ini perlu dilakukan karena sistem OCR tidak mengenali wilayah putih sehingga akan banyak wilayah teks yang tidak dikenali.

2. Ekstraksi teks pada citra

Setelah proses pengenalan wilayah teks terbentuk maka proses dilanjutkan dengan ekstraksi teks pada citra menggunakan *library tesseract*. Teks hasil ekstraksi ini menjadi data awal yang akan dilakukan proses pembersihan dan normalisasi teks agar di peroleh kumpulan *keyword*. *Keyword* ini yang akan menandai setiap citra dan menjadi fitur unik dari citra. Proses normalisasi meliputi: *case folding*, *tokenisasi*, *stopword*, dan normalisasi teks.

3. Ekstraksi fitur pada citra

Pada tahap ekstraksi fitur dilakukan dengan menggunakan algoritma SIFT dan kemudian hasil proses ekstraksi vector fitur disimpan kedalam *database*. Selanjutnya dalam pengujian fitur dari data uji akan dilakukan satu persatu dengan dibandingkan dengan fitur yang sudah tersimpan pada *database*. Proses pengenalan awal dari ekstraksi fitur citra dengan mendeteksi keypoints (titik pusat pola local) pada citra. Menentukan keypoint citra yang memiliki ketahanan terhadap perubahan dan dapat mewakili obyek citra secara keseluruhan. Hasil pembentuka pola pada citra ditandai dengan pola lingkaran pada citra. Pada gambar 2 untuk obyek citra dengan atribut nama file 2.jpg dihasilkan 462 keypoint



Gambar 2 Pembentukan Descriptor Keypoint dengan SIFT

Hasil dan Pembahasan

Pada penelitian ini akan menggunakan data citra meme dalam bahasa Indonesia untuk dijadikan *database* awal. Proses pembentukan dataset dilakukan melalui beberapa tahapan berikut ini untuk mendapatkan informasi yang tidak terstruktur menjadi informasi terstruktur. Pada penelitian ini citra yang digunakan sebagai data uji sebanyak 20 citra meme berbahasa Indonesia. Data citra terbagi menjadi 2 yaitu: 10 citra dari dataset dan 10 citra dari luar dataset.

A. Implementasi Praproses Citra

Citra hasil proses crawling yang diperoleh memiliki ukuran minimal 300 piksel x 400 piksel dan maksimal sebesar 1028 x 1028 piksel seperti pada gambar 1. Tahap pertama citra akan dirubah dari citra berwarna RGB ke citra berwarna abu-abu. Setelah diperoleh citra abu-abu proses berikutnya adalah fungsi pembuatan kernel. Fungsi ini digunakan untuk melakukan pemindaian dan konvolusi pada citra dengan transformasi morfologi. Uji coba deteksi teks dengan menggunakan operasi morfologi untuk dapat mengenali wilayah teks pada citra.

Ukuran kernel sangat mempengaruhi seberapa tebal obyek yang akan terkikis atau terlupus dengan operasi morfologi. Pada tahap ini kernel yang digunakan berukuran 3x3 dengan menggunakan bentuk elipse. Pembentukan kernel dengan ukuran tersebut nantinya digunakan untuk operasi morfologi gradien pada citra. Tahapan selanjutnya adalah melakukan proses tresholding menggunakan metode Otsu. Setelah didapatkan citra hasil, dilakukan proses pembentukan kernel berukuran 5x5 dengan menggunakan bentuk kotak. Pembentukan kernel ini digunakan untuk operasi morfologi

close, untuk melihat kontur citra yang saling terhubung. Pemilihan komposisi kernel dipilih setelah dilakukan beberapa uji coba kombinasi kernel. Komposisi kernel 3x3 untuk morfologi gradient dan kernel 5x5 untuk morfologi close.

Uji coba tahapan sistem ditunjukkan seperti pada gambar 3. Uji coba untuk citra meme dengan warna teks terang dan gambar 3 citra meme dengan warna teks gelap. Pada bagian (a) menunjukkan gambar asli sebagai *input* sistem, (b) hasil proses merubah citra RGB menjadi citra *grayscale*, (c) menunjukkan hasil proses *Otsu Thresholding*, (d) *bounding box* wilayah teks, (e) *connected region text*, (f) hasil luaran sistem.



Gambar 3 Contoh Praproses Citra

B. Implementasi Ekstraksi Teks

Setelah praproses citra telah dilakukan dan memperoleh wilayah teks yang di tandai dengan kotak bergaris hijau. Proses ekstraksi ini akan menggunakan library tesseract sebagai modul pembaca dan pengenalan citra optis (Özgen, Fasounaki, & Ekenel, 2018). Teks hasil ekstraksi ini akan menjadi data awal yang akan dilakukan proses pembersihan dan normalisasi teks agar di peroleh kumpulan *keyword* yang akan menandai setiap citra dan menjadi fitur unik dari citra. Pada gambar 5 merupakan salah satu ujicoba proses ekstraksi dimana, wilayah teks dapat dikenali secara utuh dan hasil ekstraksi juga menunjukkan hasil yang baik dengan mengenali keseluruhan teks pada citra meme. Hasil yang cukup baik ini di pengaruhi kondisi citra yang memiliki piksel yang cukup baik, warna wilayah teks yang cukup kontras dengan wilayah gambar. Proses ekstraksi fitur untuk pemrosesan teks hasil dari pengolahan OCR sehingga akan diperoleh kumpulan data *keyword* dan kemudian disimpan ke dalam *database*.



Gambar 4 Hasil deteksi teks pada citra sesuai bounding box

C. Implementasi Pos-proses Teks dan Normalisasi

Proses ini meliputi proses: *case folding*, tokenisasi, *stopword removal* dan normalisasi teks. Proses *case folding* dilakukan pada keseluruhan data hasil ekstraksi teks yang tersimpan pada file berlabel *dataframe.csv*. Dimana proses ini untuk menyamakan teks menjadi *lower case*. Proses ini menjadi langkah awal dari pengolahan normalisasi dan pembersihan teks dari *noise* yang muncul dalam proses penelitian ini. Selanjutnya tahapan tokenisasi dilakukan untuk pemotongan string input berdasarkan tiap kata yang menyusunnya.

Metode yang digunakan adalah *Term Frequency Inverse Document Frequency* (TF-IDF). Metode ini digunakan untuk melakukan pembobotan pada setiap kata yang akan digunakan pada proses temu kembali informasi, Sehingga bobot hubungan antara sebuah kata dan sebuah dokumen akan tinggi apabila frekuensi kata tersebut tinggi di dalam dokumen dan frekuensi keseluruhan dokumen yang mengandung kata tersebut yang rendah pada kumpulan dokumen (Intan et al., 2006). Pada tiap term di setiap dokumen teks hasil ekstraksi OCR dalam *corpus*. Persamaan perhitungan untuk metode TF-IDF

$$W_{dt} = TF_{dt} \times IDF_t \quad (12)$$

IDF menunjukkan ketersediaan sebuah kata *term* pada seluruh dokumen. Pada dokumen yang mengandung semakin sedikit *term* maka membuat nilai IDF semakin bernilai besar.

$$IDF_t = \log \frac{D}{df} \quad (13)$$

Dimana :

D adalah total dari keseluruhan dokumen

df adalah banyak dokumen yang memiliki term

TF adalah jumlah atau frekuensi kemunculan term t dalam dokumen d

W adalah bobot dokumen ke d terhadap kata term ke t

Tabel 1 Contoh Praproses teks

Keterangan	Teks
Hasil OCR	DIKASIH TUGAS 2 MINGGU YANG LALU, BARU DIKERJAIN PAGI-PAGI DI SEKOLAH SEBELUM DIKUMPULKANY3 & YAY L N / N GA EA ,-/ "MURID LEGEND"
Case folding	dikasih tugas 2 minggu yang lalu, baru dikerjain pagi-pagi di sekolah sebelum dikumpulkany3 & yay l n / n ga ea ,-/ "murid legend"
Tokenisasi	dikasih, tugas, 2, minggu, yang, lalu, baru, dikerjain, pagi-pagi, di, sekolah, sebelum, dikumpulkany3, &, yay, l, n, /, n. ga. Ea. -/, " , murid, legend, "
Stopword	dikasih, tugas, minggu, dikerjain, pagi-pagi, sekolah, dikumpulkany3, ga, ea, murid, legend
Normalisasi	Dikasih, tugas, minggu, dikerjain, pagi, sekolah, dikumpulkan, murid, legend

Pada Tabel 1 menunjukkan proses pengolahan teks menggunakan tahapan pra proses teks dimana, proses ini akan memberikan data teks yang sudah mengalami pembersihan dan penataan secara teratur. Kemudian setelah tahap tokenisasi akan dilakukan pengolahan dan pencarian kandidat *keyword* dengan bantuan algoritma TF IDF. Proses ini akan memberikan pembobotan pada setiap kata hasil tokenisasi sehingga sampai di peroleh kata yang memiliki nilai bobot baik. Pada proses ini diperoleh sebanyak 1657 term dari proses pembobotan kata menggunakan TFIDF. Hasil perhitungan bobot TFIDF terhadap setiap term pada keseluruhan dokumen mendapatkan term tenggelamkan memiliki nilai tertinggi karena terdapat pada 50 dokumen pada *database*.

Tabel 2 nilai pembobotan term dengan TFIDF

	Term	Rank
910	tenggelamkan	26.206278
560	mantan	21.192048
77	baru	13.726510
738	pas	12.330564
141	cinta	12.110186
...
454	kembang	0.908488
441	kehujanan	0.908488

Hasil pemringkatan pada *term* akan memudahkan dalam pencarian citra berdasarkan kata-kata tertentu, dimana hal ini akan penulis gunakan pada penelitian selanjutnya terkait dengan temu kembali informasi. Peningkatan nilai akurasi juga telah mengurangi jumlah karakter yang tidak dikenali oleh sistem.

D. Implementasi Ekstraksi Fitur Citra

Pada tahapan uji coba proses ekstraksi fitur pada citra meme menggunakan library dari OpenCV. Sehingga pembuatan deskriptor pada citra dapat dilakukan untuk mengenali pattern feature dari sebuah citra dan mencari kemiripan dari sebuah citra. Proses pengenalan awal dari ekstraksi fitur citra dengan mendeteksi keypoints (titik pusat pola local) pada citra. Setelah melewati tahapan-tahapan tersebut, hasil luaran digunakan sebagai dataset awal dalam pengembangan sistem temu kembali informasi yang di usulkan pada penelitian ini.

Dimana sistem yang di usulkan dalam penelitian ini akan menampilkan hasil *query* citra berdasarkan nilai kemiripan. Nilai kemiripan ini di ukur berdasarkan nilai kemiripan fitur citra SIFT dan nilai kemiripan teks yang menempel pada citra. Perhitungan nilai kedekatan antar vektor citra query dengan citra pada database akan menentukan akurasi dalam pengembalian informasi. Nilai kemiripan citra SIFT dihitung menggunakan persamaan *cosine distane*. Seperti pada gambar 5 menunjukkan hasil pencarian titik kunci yang memiliki kemiripan antar citra. Sedangkan untuk nilai kemiripan teks akan dilakukan dengan menggunakan perhitungan Jaro-Winkler.



Gambar 5 Uji coba pencocokan pada titik kunci pada fitur Citra

E. Implementasi Pencarian Citra Berdasarkan *Query Input*

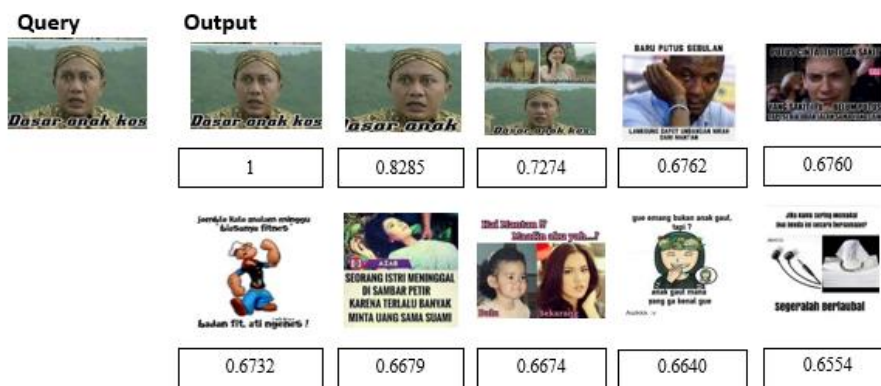
Uji coba aplikasi dalam mengenali citra *query* pada sistem yang diusulkan dalam penelitian ini dilakukan dalam beberapa tahapan. Penjelasan terkait masing-masing tahapan tersebut dijelaskan sebagai berikut:

1. Citra *query input* dilakukan proses ekstraksi fitur dengan menggunakan algoritma SIFT untuk mendapatkan fitur pada citra dalam bentuk vektor berukuran 32 sedangkan untuk *descriptor* vektornya berukuran 64. (S. Meshram & M. Agarkar, 2015)
2. Citra *query* dilakukan pencocokan dengan citra di *database* berdasarkan vektor fiturnya, dimana proses perhitungan kedekatan jarak antar vektor dari *citra query* dan citra *database* menggunakan persamaan *cosine distance*.
3. Setelah diperoleh 10 citra dengan nilai kemiripan tertinggi, kemudian dilakukan pengujian kembali untuk kemiripan pada *keyword* citra query dengan data *keyword* dari citra luaran yang tersimpan di *database*. Nilai kemiripan *keyword* ini akan menggunakan perhitungan Jaro-Winkler.
4. Proses berikutnya melakukan perhitungan rata-rata nilai kemiripan dengan menggunakan persamaan berikut

$$Sim = \frac{Nilai\ kemiripan\ cosine + Nilai\ kemiripan\ teks\ jaro}{2}$$

- Setelah proses perhitungan kemiripan kemudian dilakukan pemeringkatan dengan menampilkan 10 citra yang memiliki nilai kedekatan tertinggi dengan *citra query*.

Pengujian awal sistem dilakukan dengan mencoba melakukan proses pencarian citra dengan menggunakan *database* yang memiliki kemiripan gambar background dengan citra *query*. Hal ini dilakukan untuk mengukur akurasi sistem apakah bekerja dengan baik atau tidak. Uji coba yang saya lakukan menggunakan 50 data uji, sedangkan yang saya gunakan baru 20 data uji citra. Dengan komposisi data uji 10 citra di ambil dari *database* dan 10 citra di ambil dari luar *database*. Pada gambar 6 salah satu proses *query* sistem dalam memberikan informasi kembali dari *database* berdasarkan citra *query input*.



Gambar 6 Contoh hasil proses *query* sistem.

Penilaian kebenaran dari luaran *query* pada penelitian ini menggunakan penilaian obyektif dan penilaian subyektif. Untuk penilaian obyektif didasarkan pada nilai kecocokan yang dihasilkan dengan metode cosine dan Jaro-Winkler. Pada gambar 6 sistem memberikan luaran *query* yang memiliki kesamaan pada rentang nilai kemiripan 1 sampai dengan 0.700. Rentang nilai ini kemudian digunakan sebagai aturan dalam memberikan label benar atau salah pada hasil luaran *query*. Pemberian label benar dan salah di lakukan berdasarkan pada nilai kecocokan dengan batas minimum adalah 0.700. Jika nilai kecocokan adalah ≥ 0.700 maka akan diberi label “benar” pada hasil luaran. Tetapi jika kecocokan adalah < 0.700 antara citra *query* dengan citra dari *database* akan diberi label “salah”.

Sedangkan sebagai pembanding dilakukan penilaian subyektif didasarkan pada pengamatan langsung secara visual hasil temu kembali dari sistem oleh peneliti. Selanjutnya memberikan label benar dan salah pada hasil output sistem pencarian. Label benar di berikan jika memenuhi kriteria latar yang mirip dan teks yang mirip minimal 1 kata. Pengujian temu kembali citra disajikan dalam beberapa tabel di bawah ini. Untuk kolom pengujian subyektif merupakan hasil pengamatan dari peneliti terhadap luaran *query* dari sistem. Dalam mengukur efektifitas sebuah sistem temu kembali informasi memenuhi permintaan informasi dan kemampuan sistem menyediakan dokumen yang relevan digunakan perhitungan *precision* dan *recall*. (Dwiyantoro, 2017; Hasugian, 2006; Lestari, 2016)

$$precision = \frac{jumlah\ dokumen\ relevan\ yang\ terambil}{jumlah\ dokumen\ hasil\ pencarian}$$

Tabel 3 Hasil Nilai Precision Uji Coba query

Nilai Precision Uji Coba		Query	
		Dari Dataset	Dari Luar Dataset
20 Keypoint Match	Precision obyektif	0.32	0.05
	Precision Subyektif	0.35	0.17
40 Keypoint Match	Precision obyektif	0.36	0.12
	Precision Subyektif	0.35	0.20

Pada tabel 4 hasil pengujian dengan 20 *keypoint* kecocokan antara citra *query* dengan citra luaran dari *database*. Untuk *query* yang diambil dari dataset nilai rata-rata *precision* mencapai 32% untuk penilaian secara obyektif. Sedangkan nilai rata-rata *precision* mencapai 35% untuk penilaian secara subyektif. Untuk *query* non dataset nilai rata-rata *precision* cukup rendah untuk kedua penilaian yaitu 5% dan 17%. Karena nilai *precision* yang cukup rendah maka dilakukan pengujian kembali dengan meningkatkan jumlah *keypoint match* menjadi 40. Hasil uji coba menunjukkan ada peningkatan nilai akurasi terutama pada *query* non dataset menjadi 12% untuk penilaian obyektif dan 20% untuk penilaian subyektif. Nilai *precision* yang didapat dari keseluruhan ujicoba yang berada di bawah 50 % masih dibutuhkan pengembangan lebih lanjut terutama dalam penentuan jumlah *keypoint* yang cocok pada citra. Uji coba sistem masih perlu banyak dilakukan agar hasil pengembalian sistem dapat diperbaiki lagi.

Kesimpulan

Dari hasil rangkaian ujicoba pada penelitian ini, di dapatkan kesimpulan. Pengolahan praproses pada citra sebelum proses ekstraksi dapat membantu meningkatkan hasil pengenalan teks pada citra. Kualitas penandaan wilayah yang mengandung teks dan non teks pada citra di pengaruhi banyak factor antara lain: ukuran piksel asli dari citra itu sendiri, penggunaan warna antara teks dan latar, nilai kontras dan pencahayaan pada citra serta jenis *font* yang digunakan. Penggunaan metode SIFT dalam proses ekstraksi lokal fitur pada citra meme dapat dilakukan. Hasil fitur pola dan *keypoint* dapat digunakan dalam proses pencocokan dengan citra *query*. Pengujian menggunakan jumlah kecocokan *keypoint* dapat meningkatkan kecocokan pada hasil luaran *query*. Meskipun hasil luaran dari sistem masih di bawah hipotesa awal yaitu hanya dibawah 50%. Penggunaan metode Jaro-Winkler dalam mencocokkan antara kata pada keyword *query* dengan kata pada citra yang tersimpan di *database* dapat dilakukan dan membantu peningkatan nilai luaran dari sistem temu kembali.

Bibliografi

- Amalia, Amalia, Sharif, Arner, Haisar, Fikri, Gunawan, Dani, & Nasution, Benny B. (2018). Meme opinion categorization by using optical character recognition (OCR) and naïve bayes algorithm. *Proceedings of the 3rd International Conference on Informatics and Computing, ICIC 2018*. <https://doi.org/10.1109/IAC.2018.8780410>
- Cheriet, M., & Kharna, N., Liu, C., Suen, C. Y. (2006). *Character Recognition System A Guide for Student and Practioners*. Retrieved from https://books.google.co.id/books?id=txYpjiK_BmgC&printsec=frontcover&redir_esc=y#v=onepage&q&f=false. Di akses pada tanggal 10 oktober 2020
- Dawkins, Richard. (2016). *The Selfish Gene: 40th Anniversary Edition (4th Ed.)* (Terjemahan; Kepustakaan Populer Gramedia, ed.). PT. Gramedia.
- Dwiyantoro. (2017). Sistem Temu Kembali Informasi Dengan Keyword. *Jurnal Ilmu Perpustakaan, Informasi, Dan Kearsipan*, 5(2), 164–174.
- Hasugian, Jonner. (2006). Penelusuran Informasi Ilmiah Secara Online: Perlakuan Terhadap Seorang Pencari Informasi Sebagai Real User. *Pustaka*, 2(1), 1-13–13.
- Intan, Rolly, Defeng, Andrew, Informatika, Jurusan Teknik, Industri, Fakultas Teknologi, Kristen, Universitas, & Surabaya, Petra. (2006). Hard: Subject-Based Search Engine Menggunakan Tf-Idf Dan Jaccard'S Coefficient. *Jurnal Teknik Industri*, 8(1), 61–72. <https://doi.org/10.9744/jti.8.1.pp.61-72>.
- Kurniawati, Anna, Puspitodjati, Sulisty, & Rahman, Sazali. (2010). Implementasi Algoritma Jaro-Winkler Distance untuk Membandingkan Kesamaan Dokumen Berbahasa Indonesia. *Proceeding, Seminar Ilmiah Nasional Komputer Dan Sistem Intelijen KOMMIT 2008, Depok, Indonesia*.
- Lestari, Nisaa Putri. (2016). Uji Recall and Precision Sistem Temu Kembali. *Universitas Airlangga*, 5(3), 45–46. Retrieved from <http://journal.unair.ac.id/LN@uji-recall-and-precision-sistem-temu-kembali-informasi-opac-perpustakaan-its-surabaya-article-10825-media-136-category-8.html>
- Nakashima, Yuji, & Kuroki, Yoshimitsu. (2017). Sift feature point selection by using image segmentation. *2017 International Symposium on Intelligent Signal Processing and Communication Systems, ISPACS 2017 - Proceedings, 2018-Janua*, 275–280. <https://doi.org/10.1109/ISPACS.2017.8266488>
- Özgen, Azmi Can, Fasounaki, Mandana, & Ekenel, Hazim Kemal. (2018). Text detection in natural and computer-generated images. *26th IEEE Signal Processing and Communications Applications Conference, SIU 2018*, 1–4. <https://doi.org/10.1109/SIU.2018.8404600>

- S. Meshram, Kimaya, & M. Agarkar, Ajay. (2015). Content Based Image Retrieval Systems using SIFT: A Survey. *International Journal of Electronics and Communication Engineering*, 2(10), 18–25. <https://doi.org/10.14445/23488549/ijece-v2i10p105>
- Saggion, Horacio, & Poibeau, Thierry. (2013). *Multi-source, Multilingual Information Extraction and Summarization*. 3–21. <https://doi.org/10.1007/978-3-642-28569-1>
- Unar, Salahuddin, Wang, Xingyuan, Wang, Chunpeng, & Wang, Yu. (2019). A decisive content based image retrieval approach for feature fusion in visual and textual images. *Knowledge-Based Systems*, 179, 8–20. <https://doi.org/10.1016/j.knosys.2019.05.001>