

Pengembangan Deteksi Kebakaran Pada Urutan Video Dengan Menggunakan Metode Simple Linear Iterative Clustering

Nandy Rizaldy Najib¹⁾, Kazman Riyadi²⁾

^{1,2)} Teknik Elektro Politeknik Negeri Ujung Pandang
nandy@poliupg.ac.id¹⁾, kazmanriyadi@poliupg.ac.id²⁾



Abstrak

Di sebuah perusahaan ataupun perumahan saat ini sangat diperlukan adanya sebuah pengamanan secara dini dalam mengantisipasi bahaya kebakaran yang dapat memakan korban jiwa maupun material. Tujuan dari penelitian ini adalah membuat sistem deteksi kebakaran secara otomatis dengan kecepatan dan ketepatan yang dibutuhkan guna menghindari terjadinya kebakaran secara dini. Data yang digunakan adalah data hasil pengukuran untuk mendeteksi area bergerak menggunakan Algoritma *Gaussian Mixture Model* pada area latar belakang video. Data ini diteruskan menggunakan Metode *Simple Linear Iterative Clustering* untuk membedakan antara area api dan non-api dengan mengambil range warna RGB dan HSV api yang berbeda dibandingkan warna area disekitarnya pada video yang direkam melalui kamera. Hasil penelitian ini menunjukkan bahwa metode yang diusulkan merupakan solusi yang tepat melebihi algoritma deteksi kebakaran api pada urutan video lainnya dengan rata-rata *fps* sebesar 118,61 detik pada resolusi video 320x240 *Pixels*, dimana hasil rata-rata deteksi pada siang hari *accuracy* 97,96% dan pada malam hari *accuracy* 98,65% memberikan keandalan yang tinggi dan rendahnya tingkat kesalahan pada *alarm* yang berbunyi ketika kebakaran terdeteksi pada video.

Kata Kunci: Deteksi Kebakaran, *Gaussian Mixture Model*, *Simple Linear Iterative Clustering*, RGB, HSV.

I. PENDAHULUAN

Api merupakan sumber energi yang sangat bermanfaat bagi manusia jika dapat dikendalikan dan juga sangat penting untuk menghindarinya jika terjadi kebakaran. Seperti diketahui, kebakaran merupakan peristiwa bencana yang ditimbulkan oleh api, dimana api dapat diartikan sebagai hasil dari reaksi pembakaran yang cepat [1]. Kebakaran yang timbul terjadi karena tidak terkendalinya api yang dapat membahayakan jiwa maupun harta benda. Kebakaran dapat terjadi di hutan, bangunan di perkotaan, perumahan, tempat umum dan lain-lain baik siang maupun malam hari. Gambar 1 dan Gambar 2 secara berurutan menunjukkan kebakaran di hutan dan bangunan pada siang hari. Gambar 3 dan Gambar 4 secara berurutan menunjukkan kebakaran di hutan dan ruangan pada malam hari.

Deteksi kebakaran telah menjadi sangat menarik bagi keamanan pribadi dan aplikasi komersial, dan beberapa metode konvensional telah diusulkan untuk mendeteksi api. Namun, sebagian besar metode memerlukan jarak yang dekat dengan sumber api dan berdasarkan sensor partikel [2]. Oleh karena itu, mereka tidak bisa mendeteksi api di ruang terbuka atau besar dan tidak dapat memberikan tambahan informasi mengenai proses pembakaran. Untuk mengatasi

kelemahan ini, video deteksi kebakaran merupakan penelitian yang cocok.

Kebanyakan algoritma deteksi kebakaran ini didasarkan pada pengenalan warna pixel, deteksi gerak, atau keduanya. Dalam [3], analisis dinamis nyala api menggunakan RGB/model warna HIS digunakan untuk menentukan keberadaan api. Namun, aturan pengambilan keputusan dari metode ini adalah tidak baik di area yang membedakan api yang nyata dari area gerakan atau kebisingan karena perbedaan api diukur hanya antara dua frame berturut-turut. Dalam [4], batas api diwakili di wavelet domain, dan frekuensi tinggi sifat dari area batas api yang digunakan untuk spasial Model api yang berkelap-kelip, menghasilkan hasil yang baik. Dalam [5], metode baru deteksi nyala api yang digunakan untuk sifat warna pixel api diusulkan. Metode ini menggunakan ruang warna YCbCr karena warna RGB ruang memiliki ketergantungan pencahayaan yang lemah. Ini berarti bahwa jika pencahayaan dalam perubahan gambar, klasifikasi pixel api aturannya tidak akan dilakukan dengan baik. Dalam [6], metode pengenalan api, untuk mendeteksi secara dini kemungkinan terjadinya kebakaran secara visual dengan menggunakan tiga proses dasar yaitu: berupa deteksi warna, pencahayaan, dan gerakan. Sehingga sistem telah berhasil membedakan objek berupa api

dan objek bukan api di dalam ruangan. Beberapa algoritma di atas adalah diterapkan pada sistem nyata dengan cukup sukses. Namun, masing-masing metode ini masih memiliki aplikasi terbatas dan cukup memiliki kekurangan pada ketahanan. Oleh karena itu, untuk meningkatkan kinerja deteksi kebakaran secara efisien dalam membangun sistem yang dapat mendeteksi kebakaran pada video, metode Gaussian Mixture Model (GMM) dan Simple Linear Iterative Clustering (SLIC) yang digunakan pada sistem deteksi kebakaran dapat dilakukan dengan tingkat akurasi yang cukup tinggi.



Gambar 1. Kebakaran hutan pada siang hari



Gambar 2. Kebakaran bangunan pada siang hari



Gambar 3. Kebakaran hutan Pada malam hari



Gambar 4. Kebakaran ruangan pada siang hari

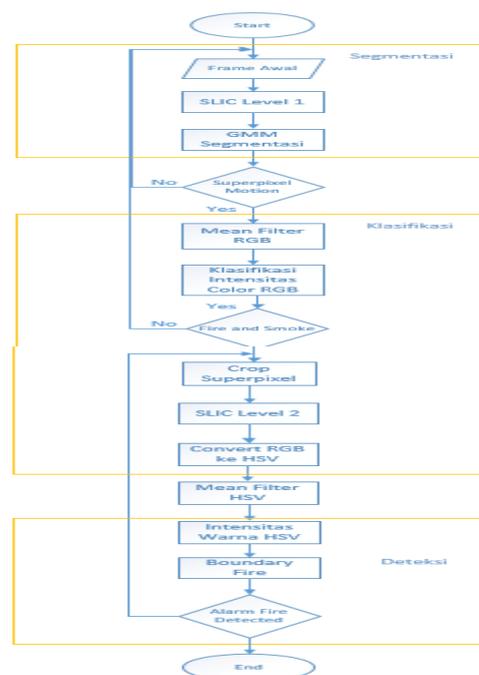
II. DASAR TEORI

2.1. Fitur Api

Fitur api memainkan peran penting dalam pengembangan sistem deteksi kebakaran karena fitur ini digunakan untuk membedakan antara api dan non-api. Dalam penelitian ini, karakteristik yang penting pada api seperti warna, area, waktu, dan ruang yang di analisis. Dalam prakteknya, kebanyakan bahan bakar akan terbakar di bawah kondisi yang sesuai, bereaksi dengan oksigen dari udara, menghasilkan produk pembakaran, memancarkan cahaya, dan melepaskan panas. Ketika api diproduksi, biasanya muncul dari lokasi yang stabil, melayang ke atas dalam cara difusi. Warnanya biasanya berkisar dari merah ke kuning dan dapat berubah putih saat suhu sangat tinggi. Ukuran, area, bentuk dan jumlah kebakaran area di gambar bervariasi dari frame ke frame yang berubah-ubah. Permukaan dan kontur area kebakaran biasanya kasar.

2.2. Algoritma Deteksi Api

Sebuah *flowchart* pendekatan deteksi api yang diusulkan digambarkan di Gambar 5. Algoritma yang diusulkan terdiri dari empat tahap: (1) deteksi area gerak yang disegmentasi warna api menggunakan *Adaptive Gaussian Mixture Model* (GMM), (2) segmentasi warna api menggunakan *Simple Linear Iterative Clustering* (SLIC) Level 1, (3) klasifikasi ke dalam api atau non-api menggunakan *Simple Linear Iterative Clustering* (SLIC) Level 2 di tahap klasifikasi ini, Metode yang digunakan dalam membedakan antara api dan bukan api dan (4) identifikasi api menggunakan Parameter range warna RGB dan HSV berdasarkan karakteristik tempo-spasial api. Pada bagian berikut, yang diusulkan algoritma deteksi kebakaran dianalisis secara rinci.



Gambar 5. Flowchart deteksi api berbasis video yang diusulkan

2.3. Gaussian Mixture Model (GMM)

Deteksi gerakan menggunakan metode *Adaptive-Gaussian Mixture Model* [7]. Metode ini mendeteksi gerakan dengan memodelkan setiap piksel ke dalam M komponen distribusi *Gaussian* serta melakukan perbaikan bobot (π), rata-rata (μ) dan standar deviasi (σ) pada masing-masing komponen *Gaussian* tersebut. Deteksi area bergerak adalah kunci penting dalam video deteksi kebakaran, yang merupakan tahap pertama dari metode yang diusulkan. Beberapa metode telah diusulkan untuk mendeteksi area bergerak, dengan background pengurangan yang

paling khas. Seperti namanya, background pengurangan adalah proses memisahkan objek foreground dari background di urutan frame video. Banyak teknik yang berbeda telah diusulkan dengan kekuatan yang berbeda dan kelemahan dalam hal kinerja dan kebutuhan komputasi.

Di antaranya metode, Gaussian mixture model (GMM) yang pertama kali diperkenalkan oleh Stuafter dan Grimson pada tahun 1999 adalah metode yang paling banyak digunakan untuk background pengurangan karena kecepatan, kesederhanaan dan kemudahan dalam pelaksanaannya [8]. Dalam metode ini, setiap pixel dimodelkan sebagai campuran distribusi Gaussian, dan setiap nilai intensitas pixel yang tidak masuk ke dalam salah satu distribusi Gaussian dimodelkan dengan ditandai sebagai foreground pixel. Dalam rangka untuk mencapai kinerja yang tinggi, saya menggunakan model campuran Adaptif Gaussian [9], Karena berhasil dalam mengelola perubahan pencahayaan dan mengurangi efek dari gerakan kecil yang berulang seperti vegetasi bergerak dan perpindahan kamera kecil.

2.4. Segmentasi Warna Api

Segmentasi warna api dilakukan pada ruang warna HSV dengan metode superpixel SLIC. Kriteria warna api antara lain memiliki warna merah-oranye-kuning dengan kecerahan yang tinggi. Dari kriteria tersebut, warna api dapat dengan mudah dikenali dari Hue, Saturation dan Value digunakan untuk mencari kecerahan warnanya. Kami menggolongkan warna pada ruang warna HSV untuk mempermudah dalam melakukan segmentasi warna api. Nilai Hue dari 0-360 dibagi menjadi 16 kelompok gradasi warna yang didapatkan dari percobaan. Gambar 6 menunjukkan pengelompokan label warna pada Hue. Nilai dari setiap label warna dari gambar 6 dan batasan kecerahan pada Saturation dan Value ditunjukkan pada Tabel 1.

Dapat dilihat dari Tabel 1 ada satu kondisi yang belum terdaftar dalam label warna yaitu Saturation ≥ 12.5% dengan Value antara 12.5% - 87.5%. Label pada kondisi tersebut mengikuti label 1 – 16 untuk nilai Hue-nya akan tetapi dengan kecerahan yang lebih gelap sehingga labelnya menjadi merah-gelap, merah-oranye gelap, dan seterusnya. Setiap piksel pada frame HSV dicocokkan dengan label warna pada Tabel 1. Untuk mendapatkan piksel warna api.



Gambar 6. Kelompok label warna pada Hue

Tabel 1. Label nilai warna HSV

Label	Nama Warna	Nilai H	Nilai S (%)	Nilai V (%)
1	Merah	0 – 10, 356 – 360	12.5 – 100.0	87.5 – 100.0
2	Merah-Oranye	11 – 20	12.5 – 100.0	87.5 – 100.0
3	Oranye	21 – 40	12.5 – 100.0	87.5 – 100.0
4	Oranye-Kuning	41 – 50	12.5 – 100.0	87.5 – 100.0
5	Kuning	51 – 65	12.5 – 100.0	87.5 – 100.0
6	Kuning-Hijau	66 – 80	12.5 – 100.0	87.5 – 100.0
7	Hijau	81 – 140	12.5 – 100.0	87.5 – 100.0
8	Hijau-Cyan	141 – 170	12.5 – 100.0	87.5 – 100.0
9	Cyan	171 – 200	12.5 – 100.0	87.5 – 100.0
10	Cyan-Biru	201 – 220	12.5 – 100.0	87.5 – 100.0
11	Biru	221 – 245	12.5 – 100.0	87.5 – 100.0
12	Biru-Magenta	246 – 280	12.5 – 100.0	87.5 – 100.0
13	Magenta	281 – 320	12.5 – 100.0	87.5 – 100.0
14	Magenta-Pink	321 – 330	12.5 – 100.0	87.5 – 100.0
15	Pink	331 – 345	12.5 – 100.0	87.5 – 100.0
16	Pink-Merah	346 – 355	12.5 – 100.0	87.5 – 100.0
17	Hitam	0 – 360	0.0 – 100.0	0.0 – 12.5
18	Abu-abu	0 – 360	0.0 – 12.5	12.5 – 87.5
19	Putih	0 – 360	0.0 – 12.5	87.5 – 100.0

2.5. Segmentasi Warna Api dengan Simple Linear Iterative Clstering (SLIC)

Simple Linear Iterative Clstering (SLIC) adalah algoritma segmentasi berbasis superpixel, unggul dalam hal ekstraksi fitur dan analisis kuantitatif, Oleh karena itu, algoritma SLIC telah diterapkan untuk segmentasi citra baru-baru ini dan sangat tepat untuk digunakan identifikasi api dimana algoritma ini memiliki keakuratan dalam metode over-segmentasi, yang memilih pusat cluster Ci dari sampel grid biasa dimana S efisien menghasilkan superpixels. S dapat digambarkan sebagai persamaan berikut.

$$S = \sqrt{N/K} \tag{1}$$

di mana N adalah jumlah pixel gambar dan K adalah jumlah yang diinginkan dan ditetapkan dari superpixels. Sementara itu, algoritma SLIC hanya mencari piksel yang sama dari masing-

masing pusat cluster piksel untuk pengelompokan dalam $2S * 2S$ yang di setting pada area di seluruh gambar, sehingga menghasilkan superpixels lebih cepat.

III. METODE PENELITIAN

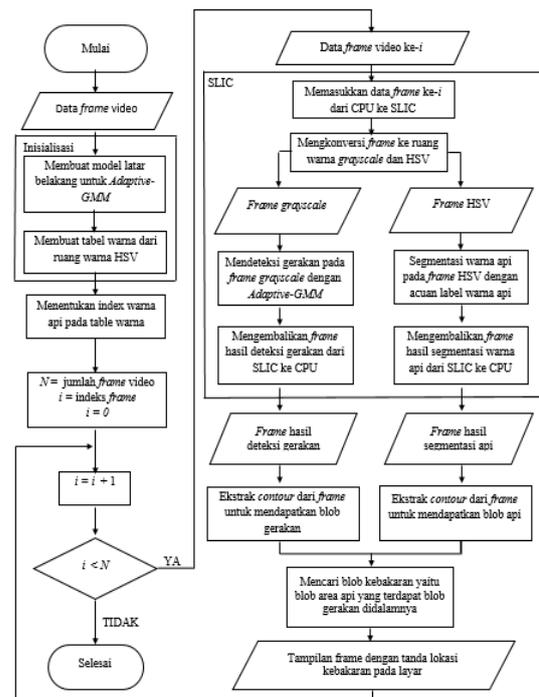
3.1. Analisis Sistem

Metode untuk mendeteksi kebakaran pada video dengan pengolahan citra sudah dikembangkan dan menunjukkan hasil yang baik diantaranya yang relevan dengan penelitian ini yaitu menggunakan deteksi gerakan yang digabungkan dengan statistik warna api untuk mendeteksi kebakaran. Pada tahap ini menggunakan *unimodal-Gaussian* untuk mendeteksi gerakan yang digabungkan dengan statistik warna api dalam ruang warna RGB untuk mendeteksi kebakaran menggunakan metode *frame different* untuk mendeteksi gerakan yang digabungkan dengan statistik warna api pada ruang warna RGB untuk mendeteksi kebakaran [10]. Deteksi area bergerak adalah kunci penting dalam video deteksi kebakaran, yang merupakan tahap pertama dari metode yang diusulkan. Beberapa metode telah diusulkan untuk mendeteksi area bergerak, dengan background pengurangan yang paling khas. Seperti namanya, background pengurangan adalah proses memisahkan objek foreground dari background di urutan frame video. Banyak teknik yang berbeda telah diusulkan dengan kekuatan yang berbeda dan kelemahan dalam hal kinerja dan kebutuhan komputasi [11]. Pada penelitian ini, kami menggunakan kombinasi antara deteksi gerakan dengan *Adaptive-Gaussian Mixture Model* dengan statistik warna api pada ruang warna RGB ke ruang warna HSV untuk mendeteksi kebakaran.

3.2. Perancangan Sistem

Secara umum, rancangan sistem deteksi kebakaran pada video berbasis pengolahan citra dengan dukungan SLIC terdiri dari pembacaan data *frame* video, inialisasi model untuk *Adaptive-GMM*, membuat tabel warna, praproses *frame* video, deteksi gerakan dengan *Adaptive-GMM*, segmentasi warna api, mendeteksi kebakaran dari kombinasi hasil deteksi gerakan dan segmentasi warna api, dan menampilkan hasil deteksi pada layar. SLIC digunakan pada metode yang memerlukan komputasi yang tinggi seperti konversi warna ke ruang warna *grayscale* dan HSV, deteksi gerakan dan segmentasi warna api. Gambar 7

menunjukkan diagram alir rancangan sistem deteksi kebakaran pada video secara umum.



Gambar 7. Rancangan sistem deteksi kebakaran pada video

Adapun penjelasan tahapan sistemnya sebagai berikut :

1. Membaca data *frame* video.
2. Inialisasi parameter dan model latar belakang untuk *Adaptive-Gaussian Mixture Model (Adaptive-GMM)*.
3. Membuat tabel label warna berdasarkan nilai warna pada ruang warna HSV untuk keperluan segmentasi warna api.
4. Memasukkan *frame* hasil pembacaan data video ke SLIC.
5. Mengkonversi *frame* dari RGB ke *grayscale* dan HSV.
6. Melakukan deteksi gerakan pada *frame grayscale* menggunakan *Adaptive-GMM*.
7. Melakukan segmentasi warna api pada *frame HSV* dengan mengacu pada tabel label warna api.
8. Mengembalikan *frame* hasil deteksi gerakan dan *frame* hasil segmentasi warna api dari SLIC ke CPU.
9. Mengekstrak *contour* pada *frame* hasil deteksi gerakan untuk mendapatkan *blob* gerakan dan pada *frame* hasil segmentasi warna api untuk mendapatkan *blob* api.

10. *Blob* hasil segmentasi warna api yang didalamnya terdapat *blob* gerakan dari hasil deteksi gerakan merupakan *blob* kebakaran.
11. Menentukan lokasi piksel kebakaran berdasarkan posisi *blob* kebakaran pada *frame*.
12. Menampilkan *frame* yang sudah ditandai lokasi kebakarannya pada layar.
13. Mengulangi langkah 4-12 sampai semua *frame* pada video diproses.

Sistem dibuat menggunakan bahasa pemrograman Matlab tambahan *library* untuk pengolahan citra yaitu OpenCV 2.4.10 yang mendukung SLIC. Spesifikasi perangkat keras menggunakan laptop dengan prosesor Intel Core i5, NVidia GT540M 2GB CUDA dan RAM 8GB. Pembacaan *frame* video, konversi warna pada SLIC, dan deteksi gerakan pada SLIC dilakukan dengan dukungan *library* OpenCV sedangkan segmentasi warna api pada SLIC menggunakan NVidia GT540M 2GB CUDA.

Compute Unified Device Architecture (CUDA) dari NVidia mengijinkan pengguna untuk menggunakan API yang tidak memerlukan pengetahuan tentang grafik *pipeline* SLIC. Kecepatan *sharing* memori diantara kelompok prosesor dapat digunakan sebagai *cache* program yang mempercepat operasi lokal. Dari pengenalan penggunaan SLIC dengan CUDA dapat digambarkan bahwa SLIC melakukan pendekatan pengolahan citra 2D sebagai bentuk dari pengolahan 3D. Secara sederhana, sebuah *quadrilateral* poligon dibariskan terhadap area kotak layar pada citra dan di-*render* ke dalam layar atau sebagai *buffer* pada memori. Pada setiap piksel dapat diterapkan pengolahan berbasis tekstur dan dapat diterapkan satu atau lebih program kecil pada piksel tersebut yang disebut dengan piksel *shader* untuk membuat warna keluaran pada setiap piksel. SLIC mengeksekusi *shader* pada banyak piksel setiap waktu secara paralel. Piksel *shader* beroperasi pada lingkungan eksekusi *IEEE floating-point* walaupun masukan dan keluaran citranya dapat diekspresikan sebagai 8-bit atau 16-bit *integer* pada setiap *channel* atau sebagai 16-bit atau 32-bit *IEEE floating-point* pada setiap *channel*. Fleksibilitas dan presisi ini mengijinkan operasi piksel untuk dieksekusi pada SLIC tanpa adanya kualitas citra yang hilang bila dibandingkan dengan metode intensif dengan CPU.

OpenCV merupakan *library* untuk pengolahan citra. Metode-metode penting dalam pengolahan citra kebanyakan sudah ada pada

OpenCV. OpenCV juga mudah untuk digunakan dan selain itu OpenCV 2.4.10 sudah mengimplementasikan beberapa metodenya pada SLIC. OpenCV menggunakan NVidia CUDA dalam implementasi metode-metodenya pada SLIC. Metode yang membutuhkan komputasi yang berat menjadi optimal pada SLIC dan didapatkan peningkatan kecepatan beberapa kali dibanding dengan CPU tergantung dari kompleksitas metodenya.

3.3. Masukan Sistem

Masukan sistem berupa data video beresolusi 320x240, 640x480 dan 800x600 yang masing-masing berjumlah 30 video dengan *fps* 30. Data video berupa hasil rekaman kebakaran dari kamera CCTV dan rekaman kamera amatir yang mempunyai kriteria berdurasi rata-rata 10 detik, rekaman bersifat statik atau relatif statik (sedikit gerakan kamera), *frame* hasil rekaman dalam ruang warna RGB, dan kebakaran dapat terlihat jelas. Data video terdiri dari kebakaran yang terjadi di hutan, bangunan, dan ruangan yang terjadi pada siang hari dan malam hari untuk variasi data.

IV. HASIL DAN PEMBAHASAN

Pengujian dilakukan pada 30 data video kebakaran yang dibagi menjadi 2 bagian yaitu kebakaran pada siang hari dan malam hari yang masing-masing sebanyak 15 sampel. Dari 15 sampel dibagi menjadi 3 lokasi yang sering terjadi kebakaran yaitu di hutan, bangunan dan di dalam ruangan. Dengan variasi sampel data video, sistem diharapkan dapat mengatasi variasi kondisi kebakaran pada waktu dan lokasi kebakaran yang berbeda. Data video untuk pengujian juga menggunakan durasi yang pendek karena untuk mencapai deteksi secara dini secepat mungkin sistem harus dapat mendeteksi kebakaran terutama pada detik-detik awal kemunculan api.

Pengujian dilakukan dengan menghitung kecepatan penerapan metode pada setiap *frame* video yang diukur dalam milidetik dan *fps* (*frame rate per second*) serta menghitung akurasi dan kesalahan deteksi kebakaran dari sampel video. Akurasi dihitung dengan membagi jumlah *frame* yang terdeteksi sebagai kebakaran dengan jumlah *frame* kebakaran sebenarnya sedangkan kesalahan deteksi dihitung dengan membagi jumlah *frame* yang terdapat *false positif* dengan jumlah *frame* kebakaran sebenarnya. *False positif* adalah jumlah *frame* yang di dalamnya ada bagian yang

bukan kebakaran tetapi terdeteksi sebagai kebakaran. *False positif* dapat dilihat dari hasil penanda lokasi kebakaran yang nampak pada layar hasil deteksi kebakaran.

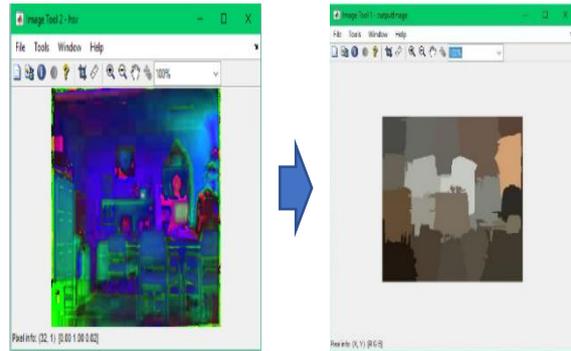
4.1. Hasil Segmentasi Api

Segmentasi warna api pada *frame* dengan melakukan pencocokan setiap piksel pada *frame* terhadap label warna pada Tabel 1 dengan menggunakan CUDA. Jumlah *thread* per blok yang digunakan yaitu 32x32 tergantung dari dukungan kartu grafis yang digunakan. Label warna api yang digunakan yaitu 1-5 yang mengacu pada Tabel 1. Gambar 8 adalah contoh video kebakaran dalam ruangan pada malam hari. Hasil segmentasi warna api Gambar 8 ditunjukkan pada Gambar 9.

Pada Gambar 8 metode dapat melingkupi api dengan baik namun refleksi api pada dinding dan lampu juga ikut tersegmentasi. Warna api pada malam hari akan nampak lebih cerah, tampak silau pada kamera, serta mempengaruhi keadaan warna disekitarnya karena cahaya disekitarnya lebih redup. Walaupun hasil dari segmentasi warna api pada malam hari tidak sebaik pada siang hari, sebagian besar kesalahan dari deteksi ini akan dapat diatasi dengan metode deteksi gerakan. Misalnya untuk cahaya lampu yang relatif statis dan pantulannya pada dinding yang juga relatif statis tidak akan di deteksi. Pada Gambar 9 segmentasi warna menggunakan ruang warna RGB yang dikonversikan ke HSV (Hue, Saturation, Value) dengan kriteria warna api sehingga metode ini hanya bisa diterapkan pada rekaman kamera yang menghasilkan *frame* RGB dan tidak bisa diterapkan pada kamera yang *frame grayscale* seperti pada mode malam.



Gambar 8. Video kebakaran di ruangan pada malam hari



Gambar 9. Hasil segmentasi warna api di ruangan pada malam hari (RGB to HSV)

4.2. Hasil Pengujian Akurasi Sistem Deteksi Kebakaran

Pengujian ini dilakukan untuk menguji akurasi dan kesalahan sistem dalam mendeteksi kebakaran. Sampel video yang diujikan 30 data video dengan resolusi 320x240 dengan 3 kelompok berdasarkan lokasi kebakaran yaitu kebakaran hutan, kebakaran di luar ruangan misalnya gedung, pom bensin, mobil, dan lain-lain serta kebakaran di dalam ruangan. Tabel 2 menunjukkan hasil perhitungan akurasi dan kesalahan deteksi kebakaran pada siang hari sedangkan Tabel 3 menunjukkan hasil perhitungan akurasi dan kesalahan deteksi kebakaran pada malam hari.

Berdasarkan Tabel 2, akurasi rata-rata yang didapatkan dari deteksi kebakaran pada siang hari yaitu 97,96% dengan kesalahan 0,25%. Kesalahan terutama disebabkan Karena pantulan cahaya api pada dinding atau obyek yang lain yang terdeteksi sebagai kebakaran. Perubahan cahaya mendadak karena kebakaran yang membesar juga menyebabkan terjadinya kesalahan deteksi. Gambar 10 menunjukkan hasil deteksi kebakaran pada data video Kebakaran kamar 1 yang ditandai dengan tanda kotak warna merah dan tulisan "*Fire*". *False positif* yang terjadi ditandai dengan tanda kotak warna merah. Pada Gambar 10 terlihat bahwa layar televisi memantulkan cahaya kebakaran sehingga sistem mendeteksinya. Kesalahan seperti ini tidak bisa dihindari oleh sistem karena dari deteksi gerakan dan warna api, keduanya menganggap layar televisi pada saat itu sebagai kebakaran.

Berdasarkan Tabel 3, akurasi deteksi kebakaran pada malam hari meningkat menjadi 98,65% karena warna api yang dominan dari sekitarnya. Akan tetapi kesalahan deteksi juga

semakin besar menjadi 0,85% dengan adanya sumber cahaya dari kebakaran menyebabkan lingkungan sekitarnya akan memantulkan cahaya tersebut.

Kemudian pada Gambar 11 menunjukkan hasil deteksi kebakaran pada data video Kebakaran ruang 3 yang ditandai dengan tanda kotak warna merah dan tulisan “Fire”. *False positif* yang terjadi ditandai dengan tanda kotak warna merah yaitu pada dinding ruangan yang memantulkan cahaya dari kebakaran. Cahaya lampu tidak terdeteksi sebagai kebakaran karena relatif statis. Karena perubahan cahaya yang mendadak dari sumber kebakaran di sekelilingnya, tanda lokasi kebakaran menjadi lebih besar. Masalah seperti ini juga tidak bisa dihindari oleh sistem karena deteksi gerakan dan warna api menganggap dinding pada saat itu sebagai piksel kebakaran. Deteksi gerakan mendeteksi gerakan dari perubahan cahaya yang mendadak karena kebakaran sedangkan segmentasi warna api mendeteksi pengaruh warna kebakaran yang memantul pada permukaan mengkilap disekelilingnya.

Tabel 2. Hasil deteksi kebakaran pada siang hari

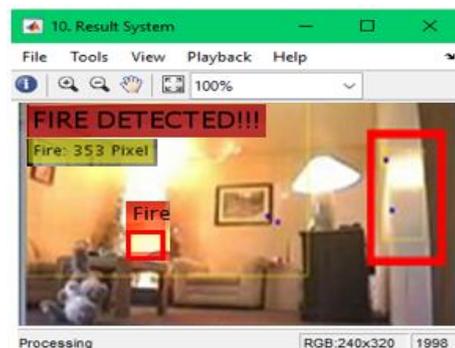
No.	Nama Video	Total Frame	Frame Kebakaran	Frame Terdeteksi (TP)	False Negatif (FN)	False Positif (FP)	Akurasi (%)	Kesalahan (%)
1	Kebakaran hutan 1	265	265	262	3	0	98,86	0,00
2	Kebakaran hutan 2	611	611	564	47	0	92,30	0,00
3	Kebakaran hutan 3	178	178	176	2	0	98,87	0,00
4	Kebakaran hutan 4	247	247	247	0	0	100,00	0,00
5	Kebakaran hutan 5	220	220	220	0	0	100,00	0,00
6	Kebakaran pom 1	542	542	542	0	0	100,00	0,00
7	Kebakaran pom 2	227	227	222	5	0	97,79	0,00
8	Kebakaran gedung 1	511	511	494	16	3	96,67	0,58
9	Kebakaran gedung 2	369	369	357	12	0	96,74	0,00
10	Kebakaran rumah	203	203	197	6	4	97,04	1,97
11	Kebakaran ruang tamu	222	222	217	5	0	97,74	0,00
12	Kebakaran dapur 1	417	417	417	0	0	100,00	0,00
13	Kebakaran dapur 2	327	327	327	0	0	100,00	0,00
14	Kebakaran kamar 1	302	302	301	1	4	99,66	1,32
15	Kebakaran kamar 2	304	304	285	0	0	93,75	0,00
	Rata-rata						97,96	0,25

Tabel 3. Hasil deteksi kebakaran pada malam hari

No.	Nama video	Total Frame	Frame Kebakaran	Frame Terdeteksi (TP)	False Negatif (FN)	False Positif (FP)	Akurasi (%)	Kesalahan (%)
1	Kebakaran hutan 1	866	866	866	0	0	100,00	0,00
2	Kebakaran hutan 2	236	236	235	1	0	99,57	0,00
3	Kebakaran hutan 3	403	403	403	0	0	100,00	0,00
4	Kebakaran hutan 4	328	328	328	0	0	100,00	0,00
5	Kebakaran hutan 5	620	620	617	3	0	99,51	0,00
6	Kebakaran pom 1	302	302	301	1	10	99,66	3,31
7	Kebakaran pom 2	297	297	270	27	9	90,91	3,03
8	Kebakaran gedung 1	937	937	937	0	0	100,00	0,00
9	Kebakaran gedung 2	164	164	159	5	0	96,95	0,00
10	Kebakaran rumah	762	762	755	7	0	99,08	0,00
11	Kebakaran ruang 1	257	257	247	10	0	96,10	0,00
12	Kebakaran ruang 2	306	306	305	1	0	99,67	0,00
13	Kebakaran ruang 3	341	341	336	5	22	98,53	6,45
14	Kebakaran ruang 4	365	365	364	1	0	99,72	0,00
15	Kebakaran ruang 5	386	386	386	0	0	100,00	0,00
	Rata-rata						98,65	0,85

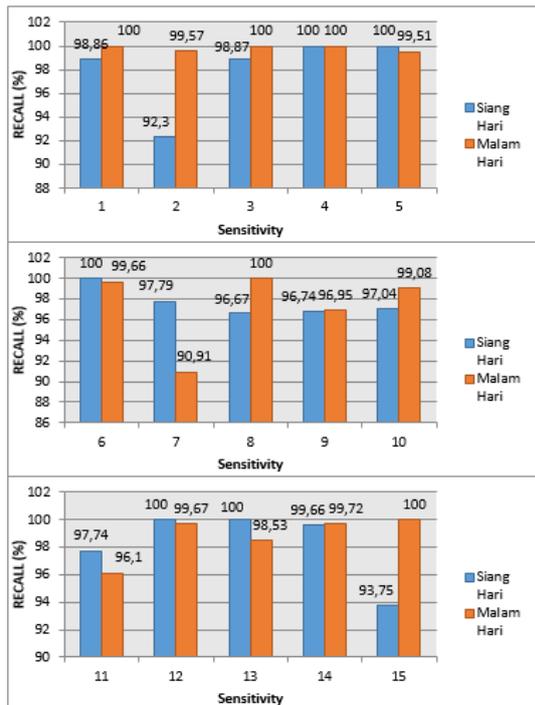


Gambar 10. Hasil deteksi kebakaran kamar pada siang hari



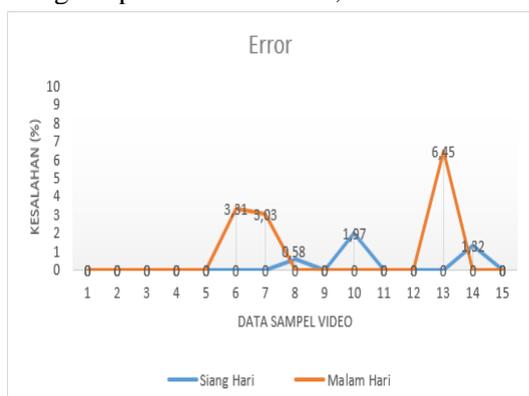
Gambar 11. Hasil deteksi kebakaran kamar pada malam hari

Berdasarkan Gambar 12 diketahui bahwa dari *recall/sensitivity* hasil Akurasi pada malam hari lebih baik daripada siang hari. Ditunjukkan dengan data sampel video pada malam hari yang sebagian besar mencapai akurasi 100%. Titik terendah akurasi pada siang hari 92,3 % sedangkan pada malam hari 90,91 %.



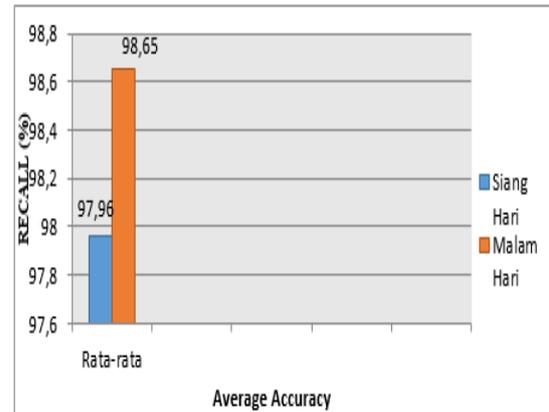
Gambar 12. Grafik Sensitivity

Berdasarkan Gambar 13 diketahui bahwa dari Error/kesalahan hasil pada siang hari lebih baik daripada malam hari. Ditunjukkan dengan data sampel video pada siang hari yang sebagian besar mencapai error 0%. Titik tertinggi error/kesalahan pada siang hari 3,31 % sedangkan pada malam hari 6,45 %.



Gambar 13. Grafik Error

Berdasarkan Gambar 14 diketahui bahwa dari hasil Accuracy rata-rata pada malam hari lebih baik daripada siang hari. Ditunjukkan dengan data sampel video pada malam hari rata-rata akurasi deteksi 98,65% dari 15 sampel video sedangkan pada siang hari 97,96 % dari 15 sampel video.



Gambar 14. Grafik Average Accuracy

4.3. Hasil Pengujian Kecepatan Sistem Deteksi Kebakaran

Pengujian ini dilakukan untuk menguji performa metode yang menggunakan SLIC dibandingkan dengan CPU. Kecepatan diukur dalam milidetik dan jumlah *frame* yang dieksekusi per detik (*fps*). Pada setiap kategori resolusi digunakan 3 video dengan *fps* 30 berdurasi rata-rata 10 detik untuk diukur kecepatan eksekusi metodenya selama 5 kali percobaan. Tabel 4 menunjukkan perbandingan kecepatan rata-rata eksekusi metode yang menggunakan CPU dan SLIC pada sampel video dengan variasi resolusi video.

Dari Tabel 4 dapat dilihat bahwa pemrosesan metode deteksi api pada video pada SLIC lebih cepat daripada CPU. Pada resolusi 800x600 pemrosesan video pada SLIC masih mendekati *fps* video aslinya sedangkan pada CPU pemrosesan video sudah jauh lebih lambat dari *fps* video aslinya. Peningkatan kecepatan pemrosesan sistem dengan SLIC sekitar 1,5 kali dari CPU. Pada metode deteksi gerakan dengan SLIC peningkatan kecepatan sekitar 3 kali dari CPU sedangkan untuk segmentasi warna api menggunakan SLIC peningkatan kecepatan sekitar 1,7 kali dari CPU. Metode yang membutuhkan pemrosesan setiap piksel pada *frame* lebih cepat apabila diproses pada SLIC yang mendukung komputasi paralel. Semakin besar resolusi video maka semakin banyak piksel yang harus diproses untuk itu komputasi paralel sangat efektif. Dengan menggunakan SLIC, sistem menjadi lebih fleksibel dan dapat memenuhi performa untuk deteksi kebakaran pada video.

Tabel 4. Kecepatan eksekusi metode dengan CPU dan SLIC

Metode	Resolusi Video	Kecepatan rata-rata (milidetik)		fps rata-rata		Peningkatan kecepatan
		CPU	SLIC	CPU	SLIC	
Deteksi gerakan (GMM)	320x240	5,33	1,49	190,4	753	3,97
	640x480	13,83	3,9	78,25	258,75	3,31
	800x600	34,52	6,63	40,67	152,3	3,75
Segmentasi warna api	320x240	3,65	1,98	274,2	536,26	1,96
	640x480	10,47	5,73	103,8	176,68	1,7
	800x600	25,82	7,55	54,91	133,45	2,44
Total metode deteksi kebakaran	320x240	13,43	8,67	75,94	118,61	1,57
	640x480	36,23	21,33	29,85	47,46	1,6
	800x600	83,66	29,22	16,17	35,46	2,19

V. KESIMPULAN

Berdasarkan hasil pengujian sistem deteksi kebakaran pada video dapat diambil beberapa kesimpulan yaitu:

1. Hasil rata-rata akurasi sistem deteksi kebakaran pada siang hari yaitu 97,96% dengan kesalahan rata-rata sebesar 0,25% dari 15 sampel video.
2. Hasil rata-rata akurasi sistem deteksi kebakaran pada malam hari yaitu 98,65% dengan kesalahan rata-rata sebesar 0,85% dari 15 sampel video.
3. Kesalahan deteksi terutama disebabkan oleh pantulan cahaya kebakaran pada permukaan yang mengkilap misalnya pada sampel kebakaran ruang tamu yang dindingnya memantulkan cahaya dari kebakaran. Adapun pengembangan yang diharapkan pada aplikasi ini yaitu penambahan pengenalan asap untuk sistem mendeteksi dini akan munculnya api dan dapat bekerja secara *real-time*.
4. Sistem mampu mendeteksi kebakaran pada video yaitu dengan rata-rata *fps* sebesar 118,61 pada resolusi 320x240, 47,46 pada resolusi 640x480, dan 35,46 pada resolusi video 800x600.
5. Peningkatan kecepatan sistem deteksi kebakaran dengan SLIC sekitar 1,5 kali dari CPU.

UCAPAN TERIMA KASIH

Pada kesempatan ini penulis menyampaikan terima kasih kepada seluruh pihak terutama kepada Pimpinan Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang yang telah memberikan dorongan dan kontribusi yang positif bagi penyelesaian penelitian ini. Penulis berharap semoga ke depannya penelitian ini

dapat berguna, bermanfaat dan dijadikan referensi bagi yang tertarik untuk mengembangkan penelitian ini.

REFERENSI

- [1] Pusdiklatkar. (2006). *Modul Pelatihan: Perilaku Api*. Jakarta: Pusdiklatkar.
- [2] Jones, W.W., 2004. An Algorithm for Fast and Reliable Fire Detection. In:8th Fire Suppression and Detection Research Application Symposium,pp.1–3.
- [3] Chen, T.H., Wu, P.H., Chiou, Y.C., 2004. An Early Fire-Detection Method based on Image Processing. In: IEEE. Int. Conf. on Image Processing, vol.3 ,pp.1707– 1710.
- [4] Toreyin, B.U., Dedeoglu, Y., Gudukbay, U., Centin, A.E., 2006. Computer vision-based method for real-time fire and flame detection. *Pattern Recog. Lett.* 27, 49–58.
- [5] Celik, T., Demirel, H., 2009. Fire detection in video sequences using a generic color model. *Fire Saf. J.* 44, 147–158.
- [6] Shidik G.F. (2013). *Sistem Deteksi Dini Kebakaran Berbasis Camera Surveillance dengan Metode Back Propagation Neural Network*. Semarang: Teknik Informatika Universitas Dian Nuswantoro.
- [7] Zivkovic, Z., 2004, Improved Adaptive Gaussian Mixture Model for Background Subtraction, *Proceedings International Conference in Pattern Recognition (ICPR)*, UK, Agustus 2004.
- [8] Stauffer, C., Grimson, W.E.L., 2000. Learning patterns of activity using real-time tracking. *IEEE Trans. Pattern Anal. Machines Intel.*22, 747–757.
- [9] KaewTraKulPong, P., Bowden, R., 2002. An Improved Adaptive Background Mixture Model for Real-Time Tracking with Shadow Detection. In: Proc. 2nd European Workshop on Advanced Video-Based Surveillance System, AVBS01, pp. 135–144.
- [10] Permana A.S., Usman K., & Ary M.M. (2009). *Deteksi Kebakaran Berbasis Kamera Secara Realtime Dengan Pengolahan Citra Digital*. Bandung: Teknik Elektro Institut Teknologi Telkom.
- [11] Piccardi, M., 2004. Background Subtraction Techniques: A Review. In: IEEE. Int. Conf. on Systems. Manand Cybernetics, pp.3099–3104.