

Deteksi Aktivitas Mencurigakan Peserta *Computer Based Test* Menggunakan IP Camera

(*Suspicious Activity Detection for Computer Based Test by Using IP Camera*)

Muhammad Ihsan Zul^{[1]*}, Dzaky Kurniawan^[2], Rahmat Suhatman^[3]

^{[1][2][3]}Teknik Informatika

Politeknik Caltex Riau, Pekanbaru, Indonesia

E-mail: [ihsan, rahmat]@pcr.ac.id, dzakyk@alumni.pcr.ac.id

KEYWORDS:

Suspicious Activities, IP Camera, Feature Extraction, K-Nearest Neighbor

ABSTRACT

Common surveillance device that used to monitor an area is known as CCTV. The CCTV will provide results in the form of video recordings, which can then be accessed by wireless communication. In its use, CCTV needs humans to monitor the real condition of the area/place. Then the use of CCTV becomes less efficient when used to oversee a place where the room rarely has movement. Because CCTV cannot detect or identify suspicious actions automatically. This research aim to develop a method that can be used to identify the activity (irregular movements) automatically. In this case, the change to be determined was the activities towards the Politeknik Caltex Riau Computer Based Test (CBT) participants. The CBT room has been employed by the IP Camera to identify participant activities. The IP camera captures the image and the image is then processed by the feature extraction method. Proposed feature extraction method are background subtraction and pixel mapping. Pixel mapping is a method that maps objects based on specified ratio data. There are 18 ratio data generated by this feature extraction process. The determination of the illegal activities done by using the k-Nearest Neighbor. The Algorithm detects the illegal movement by using 502 datasets, and the accuracy obtained was between 98% - 98.4% with an average accuracy of 98.2% for the value of neighborliness = 3. The result can conclude that the method can identify the illegal activities of a CBT participant in the CBT room.

KATA KUNCI:

Aktivitas Mencurigakan, IP Camera, Ekstraksi Ciri, K-Nearest Neighbor

ABSTRAK

Perangkat pengawasan yang sering digunakan adalah CCTV. CCTV ini akan memberikan hasil berupa rekaman video yang kemudian dapat diakses dengan komunikasi wireless. Dalam pemanfaatannya, CCTV tetap harus dijaga oleh sumber daya manusia untuk memberikan peringatan. Hal ini menyebabkan tidak efisiennya penggunaan CCTV, apalagi ketika digunakan untuk mengawasi sebuah ruangan jarang digunakan. Hal ini terjadi karena CCTV tidak dapat mendeteksi atau mengidentifikasi aktivitas mencurigakan secara otomatis. Sehingga dibutuhkan metode yang dapat mengidentifikasi aktivitas secara otomatis. Penelitian ini mengajukan pengembangan metode untuk mengidentifikasi aktivitas. Aktivitas yang akan diidentifikasi adalah aktivitas peserta ujian berbasis komputer (CBT) di Politeknik Caltex Riau. Ruang CBT telah dilengkapi dengan kamera IP untuk mengidentifikasi aktivitas peserta. Kamera IP menangkap citra dan citra tersebut kemudian diproses metode ekstraksi ciri background subtraction dan pemetaan piksel. Pemetaan piksel merupakan suatu metode yang memetakan objek berdasarkan data rasio yang ditentukan. Terdapat 18 data rasio yang dihasilkan oleh proses ekstraksi ciri ini. Kemudian penentuan aktivitas yang mencurigakan dilakukan dengan menggunakan algoritma k-nearest neighbor. Algoritma mendeteksi aktivitas mencurigakan dengan menggunakan 502 datasets. Tingkat akurasi yang didapatkan adalah antara 98% - 98,4% dengan rata-rata akurasi sebesar 98,2% untuk nilai ketetanggaan = 3. Berdasarkan capaian akurasi tersebut dapat disimpulkan bahwa metode ini dapat mendeteksi aktivitas mencurigakan peserta ujian berbasis komputer di ruang CBT.

I. PENDAHULUAN

CCTV merupakan perangkat kamera yang digunakan untuk pengawasan. Hasil rekaman kamera CCTV berupa rekaman video yang disimpan

di dalam perangkat decoder atau hardisk. Akibatnya rekaman ini membutuhkan media penyimpanan dengan kapasitas yang cukup besar. Selain itu, agar proses pemantauan berjalan efektif, CCTV

membutuhkan manusia untuk memantau kejadian yang terjadi di area yang diawasi, sehingga informasi bisa didapatkan dengan cepat. Beberapa CCTV tidak dapat mendeteksi atau mengidentifikasi gerakan objek yang dipantau. Sehingga penggunaan CCTV ini hanya digunakan untuk memantau tanpa memiliki kemampuan lebih, khususnya dalam mengidentifikasi gerakan objek. Pengidentifikasian objek dibutuhkan untuk mengembangkan sistem pemantauan cerdas. Sistem yang dapat mengingatkan penggunaannya untuk mengenali gerakan-gerakan yang mencurigakan. Sehingga kemampuan kamera pengawas perlu ditingkatkan untuk mengidentifikasi dan mengenali aktivitas secara otomatis.

Keterbatasan kamera CCTV dapat diselesaikan oleh IP Camera. Saat ini, IP Camera merupakan salah satu kamera pemantau yang populer digunakan untuk mengawasi suatu area atau objek. Sehingga sejak IP Camera diperkenalkan, kemampuan IP kamera jauh di atas kamera CCTV yang masih menggunakan data analog. IP Camera memanfaatkan citra digital dalam proses streaming data. IP Camera juga memiliki kemampuan dalam berkomunikasi dengan internet, karena telah dilengkapi dengan *Internet Protocol*. Sehingga IP Camera merupakan pilihan yang lebih baik dalam pemantauan. Akan tetapi, beberapa IP kamera belum dilengkapi dengan kemampuan dalam pendeteksian objek. Kebanyakan IP Camera sudah menggunakan fitur deteksi gerak namun tidak adaptif dengan perubahan lingkungan. Pada umumnya IP Camera belum memiliki fitur yang dimanfaatkan untuk pendeteksian dan identifikasi aktivitas manusia.

Widyawan [1] menggunakan IP Camera dalam mendeteksi pergerakan objek secara dinamis dan adaptif. Penelitian ini menghasilkan algoritma yang mampu menangani perubahan kondisi area yang ditangkap. Sehingga mampu memisahkan antara objek dan latar belakang (*background*) dengan baik. Namun, penelitian tersebut tidak mendeteksi dan mengidentifikasi pergerakan yang dilakukan oleh suatu objek, khususnya manusia.

Penelitian yang dilakukan Zul [2], menggunakan IP Camera dalam pendeteksian aktivitas manusia. Aktivitas yang dikenali antara lain, berdiri, duduk, jongkok dan telentang. Zul mengembangkan metode pendeteksian aktivitas dengan menerapkan *background subtraction*. Pendeteksian ini terbatas pada jenis gerak yang akan dikenali. Sehingga

belum mengidentifikasi aktivitas tersebut secara dalam, seperti mencurigakan atau tidak mencurigakan.

Kedua penelitian yang dijabarkan pada paragraf sebelumnya jelas memiliki keterbatasan dalam mendeteksi aktivitas yang mencurigakan. Sehingga pemanfaatan hasil kedua penelitian tersebut belum sampai dalam tahap analisis aktivitas secara lebih mendalam. Penelitian ini mengajukan metode yang digunakan untuk mendeteksi aktivitas mencurigakan. Aktivitas mencurigakan pada umumnya perlu diidentifikasi di tempat-tempat tertentu yang membutuhkan pengawasan. Seperti bank, museum, dan area-area strategis yang membutuhkan pengawasan tinggi. Di dunia pendidikan, pengawasan juga dibutuhkan saat pelaksanaan ujian atau tes. Saat ini Ujian Berbasis Komputer (UBK) atau *Computer Based Test* (CBT) telah diterapkan untuk pelaksanaan test/ujian. Hal ini sejalan dengan berkembangnya teknologi yang dimanfaatkan di dunia pendidikan.

Politeknik Caltex Riau (PCR) telah menggunakan CBT dalam penerimaan mahasiswa baru. Setiap tahun kegiatan ini diikuti oleh lebih dari 1500 calon mahasiswa. Sehingga penelitian pendeteksian aktivitas mencurigakan relevan diterapkan di ruang CBT PCR. Penelitian ini diarahkan untuk mendeteksi aktivitas mencurigakan peserta CBT ketika mengerjakan soal pennerimaan mahasiswa baru.

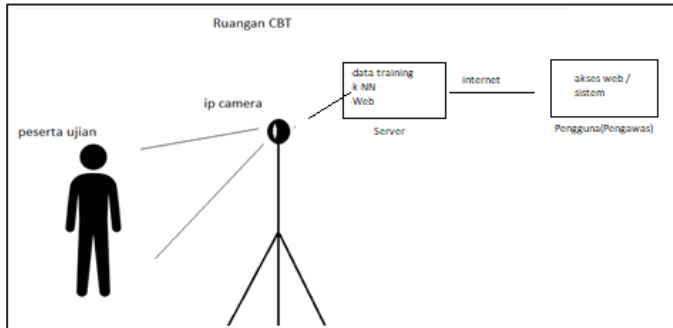
Penelitian ini melakukan pengembangan metode untuk mengidentifikasi aktivitas mencurigakan peserta ujian CBT PCR dengan menggunakan IP Camera. Proses identifikasi aktivitas mencurigakan dilakukan dengan menerapkan *background subtraction* dan pemetaan piksel untuk ekstraksi ciri objek yang akan dikenali. Pendeteksian dilakukan dengan menggunakan algoritma k-Nearest Neighbor. Pengujian kehandalan hasil deteksi dilakukan dengan mengevaluasi data training yang dihasilkan. Evaluasi dilakukan dengan menggunakan k-fold cross validation.

II. METODOLOGI

A. Arsitektur Sistem

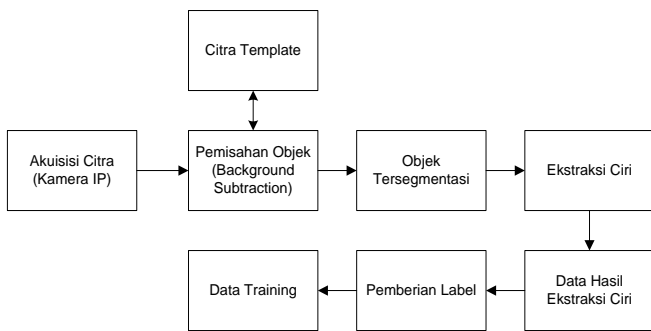
IP Camera berfungsi untuk menangkap citra secara kontinu untuk selanjutnya dapat diproses menjadi suatu citra digital. Kemudian server berfungsi sebagai tempat pemrosesan suatu citra,

membandingkan citra testing dengan citra training, dan mengidentifikasi pola aktivitas peserta ujian. Selanjutnya pengguna (pengawas) dapat mengakses web / sistem pada ruangan yang berbeda melalui internet. Gbr. 1 merupakan gambaran umum arsitektur sistem.



Gbr. 1 Arsitektur Sistem

B. Perancangan Sistem



Gbr. 2 Proses Training Sistem

Gbr. 2 merupakan proses training yang dikembangkan. Proses digunakan agar sistem dapat mengenali pola aktivitas mencurigakan dari beberapa citra training yang disimpan dalam basis data. Pemrosesan ini membutuhkan tahap-tahap hingga menghasilkan data training yang memiliki akurasi pendeteksian tinggi. Proses yang dilewati antara lain:

1) Akuisisi Citra

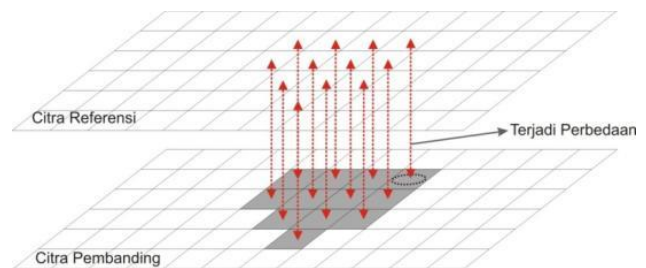
Akuisisi citra merupakan proses pertama kali yang dilakukan untuk mendapatkan citra digital. Citra digital yang dihasilkan berformat *.jpg. Citra ini ditangkap dengan mengembangkan sistem pengakuisis citra dengan mengakses IP Camera. Akuisisi dilakukan secara otomatis oleh sistem.



Gbr. 3 Contoh hasil akuisisi citra

2) Preprocessing/Praproses

Pada tahap ini bertujuan untuk memudahkan dalam proses ekstraksi ciri. Pada tahap ini dilakukan proses pemisahan objek dengan latar belakang atau background, dimana metode yang digunakan adalah *Background Substraction*. Penggunaan metode ini dapat dilihat pada referensi [3] - [5]. Metode ini dapat bekerja jika dilakukan penetapan citra latar yang akan digunakan. Selanjutnya citra yang baru ditangkap akan dibandingkan dengan citra latar dengan citra objek yang diidentifikasi sebagai objek terpilih.



Gbr. 4 Metode *Background Substraction*

Pemisahan objek dengan latar belakang juga dilakukan dengan mengatur nilai *threshold*. *Threshold* merupakan nilai toleransi yang digunakan untuk mendefinisikan kesamaan nilai RGB suatu piksel citra pada citra latar dengan objek. Perbandingan citra dengan menggunakan nilai *threshold* dilakukan dengan menggunakan rata-rata nilai RGB setiap piksel pada citra [1].

$$g_o(x, y) = \frac{g_R(x, y) + g_G(x, y) + g_B(x, y)}{3} \quad (1)$$

$$f_o(x, y) = \frac{f_R(x, y) + f_G(x, y) + f_B(x, y)}{3} \quad (2)$$

$$(g_o(x, y) - T) \leq f_o(x, y) \leq (g_o(x, y) + T) \quad (3)$$



Gbr. 5 Citra yang dijadikan sebagai *template/background*



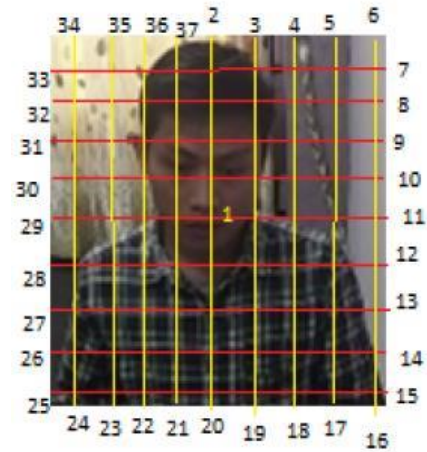
Gbr. 6 Contoh citra aktivitas



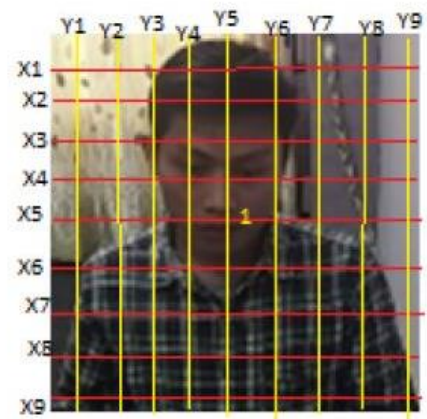
Gbr. 7 Citra yang sudah tersegmentasi

3) Ekstraksi Ciri

Proses ekstraksi ciri dilakukan untuk mencari pixel objek yang telah dipisahkan. Metode ini kemudian disebut sebagai metode pemetaan piksel. Ciri setiap objek ditentukan berdasarkan 9 data dari rasio lebar dan tinggi yaitu, $X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8, X_9, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7, Y_8, Y_9$ yang nantinya akan dijadikan sebagai atribut pada data training dalam pengklasifikasian menggunakan algoritma k-NN. Berikut ini adalah contoh citra yang dilakukan proses ekstraksi ciri.



(a)



(b)

Gbr. 8 (a) titik pada objek, (b) pembagian area objek

Titik yang telah didefinisikan dihitung rasionya dengan cara membandingkan nilai tersebut ke nilai acuan. Perhitungan nilai ini dapat dilihat pada persamaan (5)-(22). Variabel yang digunakan sebanyak 18 data, yaitu 9 rasio lebar berupa $L_1, L_2, L_3, L_4, L_5, L_6, L_7, L_8, L_9$ dan 9 rasio tinggi berupa $T_1, T_2, T_3, T_4, T_5, T_6, T_7, T_8, T_9$. Titik yang telah diidentifikasi dihitung dengan persamaan berikut:

$$T(x,y) = \frac{\min(x,y) + \max(x,y)}{2} \quad (4)$$

$$X_1 = \frac{d(\text{titik}_{33}, \text{titik}_7)}{d(\text{titik}_2, \text{titik}_{20})} \quad (5)$$

$$X_2 = \frac{d(\text{titik}_{32}, \text{titik}_8)}{d(\text{titik}_2, \text{titik}_{20})} \quad (6)$$

$$X_3 = \frac{d(\text{titik}_{31}, \text{titik}_9)}{d(\text{titik}_2, \text{titik}_{20})} \quad (7)$$

$$X4 = \frac{d(\text{titik}_{30}, \text{titik}_{10})}{d(\text{titik}_2, \text{titik}_{20})} \quad (8)$$

$$X5 = \frac{d(\text{titik}_{29}, \text{titik}_{11})}{d(\text{titik}_2, \text{titik}_{20})} \quad (9)$$

$$X6 = \frac{d(\text{titik}_{28}, \text{titik}_{12})}{d(\text{titik}_2, \text{titik}_{20})} \quad (10)$$

$$X7 = \frac{d(\text{titik}_{27}, \text{titik}_{13})}{d(\text{titik}_2, \text{titik}_{20})} \quad (11)$$

$$X8 = \frac{d(\text{titik}_{26}, \text{titik}_{14})}{d(\text{titik}_2, \text{titik}_{20})} \quad (12)$$

$$X9 = \frac{d(\text{titik}_{25}, \text{titik}_{15})}{d(\text{titik}_2, \text{titik}_{20})} \quad (13)$$

$$Y1 = \frac{d(\text{titik}_{34}, \text{titik}_{24})}{d(\text{titik}_2, \text{titik}_{20})} \quad (14)$$

$$Y2 = \frac{d(\text{titik}_{35}, \text{titik}_{23})}{d(\text{titik}_2, \text{titik}_{20})} \quad (15)$$

$$Y3 = \frac{d(\text{titik}_{36}, \text{titik}_{22})}{d(\text{titik}_2, \text{titik}_{20})} \quad (16)$$

$$Y4 = \frac{d(\text{titik}_{37}, \text{titik}_{21})}{d(\text{titik}_2, \text{titik}_{20})} \quad (17)$$

$$Y5 = \frac{d(\text{titik}_2, \text{titik}_{20})}{d(\text{titik}_2, \text{titik}_{20})} \quad (18)$$

$$Y6 = \frac{d(\text{titik}_3, \text{titik}_{19})}{d(\text{titik}_2, \text{titik}_{20})} \quad (19)$$

$$Y7 = \frac{d(\text{titik}_4, \text{titik}_{18})}{d(\text{titik}_2, \text{titik}_{20})} \quad (20)$$

$$Y8 = \frac{d(\text{titik}_5, \text{titik}_{17})}{d(\text{titik}_2, \text{titik}_{20})} \quad (21)$$

$$Y9 = \frac{d(\text{titik}_6, \text{titik}_{16})}{d(\text{titik}_2, \text{titik}_{20})} \quad (22)$$

4) Proses Testing

Terdapat dua tahapan yang sama yang digunakan untuk *testing* dan *training*. Tahapan tersebut adalah Akusisi Citra dan Ekstraksi Ciri. Kedua tahapan ini juga diterapkan pada saat *testing*. Akan tetapi pada tahap testing diterapkan algoritma yang digunakan untuk mendeteksi

aktivitas berdasarkan data training. Algoritma ini termasuk ke dalam kelompok Supervised Learning atau Classification [6]. Algoritma yang digunakan dalam pendeteksian ini adalah *k-nearest neighbor* atau k-NN.

5) Klasifikasi k-NN

Algoritma k-NN bekerja dengan mengukur jarak setiap data *training* ke setiap data *testing*. Pengukuran jarak ini dilakukan dengan menggunakan persamaan *Euclidean Distance* (ED) [6]. Data akan diklasifikasikan berdasarkan jarak terdekat ke sebanyak *k* data training. *K* adalah variabel yang ditentukan. *K* akan diuji untuk menentukan akurasi pendeteksian terbaik.

C. Pengujian

Pengujian penelitian ini dilakukan dengan mengevaluasi data training dengan menggunakan *k-fold cross validation* [6]. Metode ini dilakukan dengan menguji data training yang dihasilkan ke dirinya sendiri dengan menggunakan algoritma k-NN. Pengujian ini juga dilakukan untuk mendapatkan nilai *threshold* terbaik yang digunakan untuk background subtraction, serta memilih nilai *k* terbaik untuk algoritma k-NN.

III. HASIL DAN PEMBAHASAN

A. Hasil Pengumpulan Data Training

Pengumpulan data dilakukan dengan melakukan akusisi citra aktivitas peserta ujian secara random. Citra digital yang dihasilkan berformat *.jpg dengan ukuran 256x144 piksel. Pengambilan citra berada di dalam suatu ruangan yang sudah dilakukan pengaturan sebelumnya.



Gbr. 9 Proses Akusisi Citra

Kemudian dilakukan proses pemisahan objek dengan latar belakang.



Gbr. 10 Latar Belakang atau background

Setelah itu citra aktivitas akan diberi label, dimana pada penelitian ini label akan diberikan kadalam dua label, yaitu aktivitas mencurigakan atau tidak mencurigakan. Berikut ini hasil pengumpulan data training system.



Gbr. 11 Aktivitas Mencurigakan



Gbr. 12 Aktivitas Tidak Mencurigakan

Gbr. 14 dan Gbr.15 merupakan hasil pengumpulan data *training* yang kemudian akan disimpan ke dalam *database*. Gbr. 14 merupakan citra yang ditangkap dan diidentifikasi sebagai gerakan mencurigakan. Karena objek yang diawasi menoleh ke area lain yang dilarang pada saat pelaksanaan

CBT. Sedangkan Gbr. 15 merupakan objek yang tidak melakukan aktivitas yang dilarang pada saat pelaksanaan CBT. Sehingga diidentifikasi sebagai kegiatan yang tidak mencurigakan.

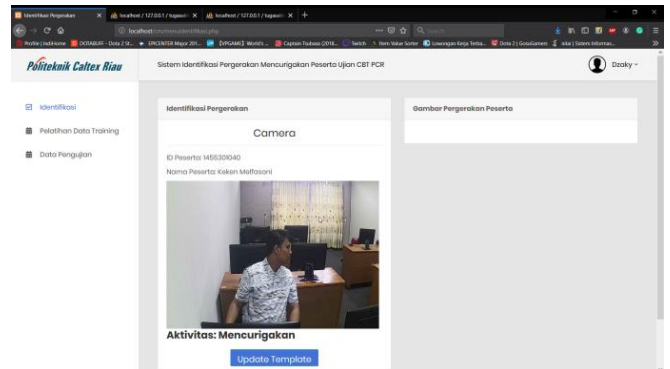
Gbr. 14 dan Gbr. 15 kemudian diekstraksi cirinya. Setiap citra akan diproses untuk mendapatkan data rasio melalui metode pemetaan piksel. Data *training* yang dihasilkan dapat dilihat pada Tabel I.

TABEL I. DATA TRAINING

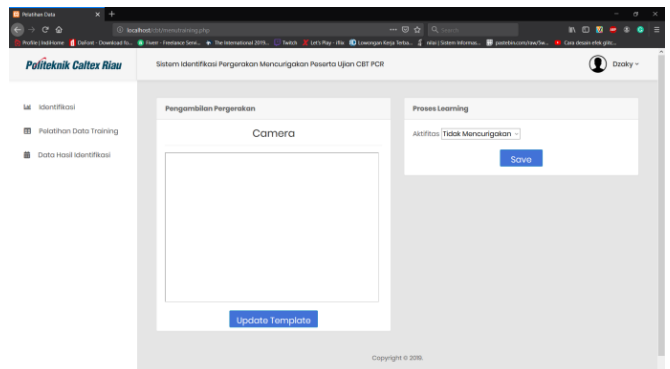
Aktivitas	X1	X2	X3	X..	Y1	Y2	Y3	Y..
TM	1,24	0,18	0,31	...	1,17	0,12	0,3	...
TM	1,23	0,43	0,22	...	1,17	0,13	0,29	...
M	1,23	0,47	0,39	...	1,16	0,13	0,31	...
TM	1,24	0,35	0,33	...	1,15	0,13	0,31	...
M	1,22	0,45	0,22	...	1,16	0,12	0,3	...
TM	1,23	0,2	0,5	...	1,17	0,13	0,24	...
M	1,21	0,22	0,57	...	1,17	0,15	0,25	...
M	1,22	0,44	0,27	...	1,17	0,13	0,3	...
TM	1,22	0,22	0,79	...	1,18	0,15	0,26	...
M	124	0,41	0,5	...	1,17	0,14	0,31	...
M	1,25	0,45	0,34	...	1,16	0,12	0,29	...

B. Hasil Perancangan Antarmuka

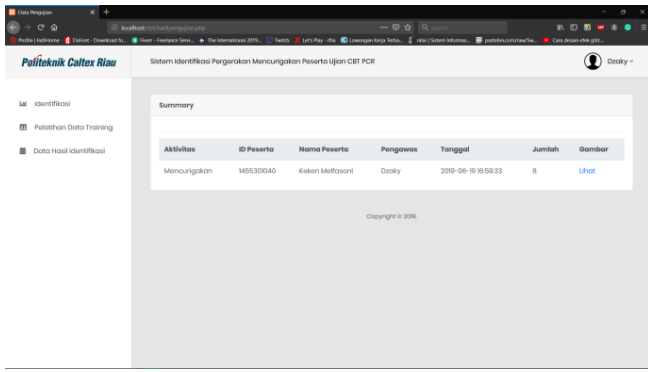
Antarmuka dirancang dengan menggunakan framework bootstrap. Antarmuka sistem mengakomodir informasi secara langsung tentang aktivitas CBT yang sedang dipantau.



Gbr. 13 Halaman Menu Identifikasi



Gbr. 14 Halaman Menu Training



Gbr. 25 Halaman Menu Data Pengujian

Gbr. 14, 15, dan 16 merupakan beberapa hasil perancangan system identifikasi aktivitas mencurigakan peserta ujian CBT Politeknik Caltex Riau yang dibangun berbasis web, dimana pada Gbr. 14 adalah halaman menu identifikasi. Halaman ini akan memproses atau melakukan proses identifikasi terhadap aktivitas peserta ujian. Pada Gbr. 15 adalah halaman menu training, dimana pada menu ini pengguna dapat melakukan penambahan data *training system*. Gbr. 16 adalah halaman menu data pengujian, dimana pada menu ini merupakan *summary*, yang akan menampilkan data hasil identifikasi aktivitas peserta ujian yang teridentifikasi sebagai aktivitas mencurigakan.

Gambaran umum hasil pengujian dapat dilihat pada Tabel II.

TABEL II. HASIL PENGUJIAN

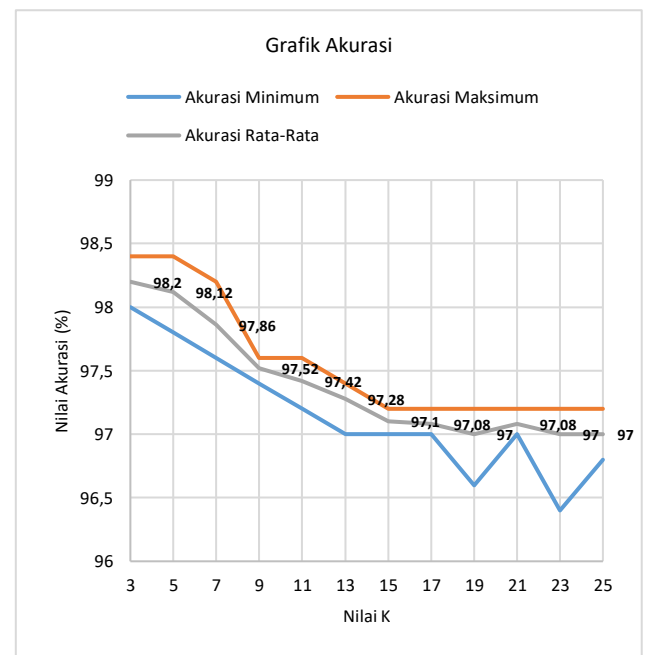
Nilai K	Akurasi (%) 10 kali percobaan					Akurasi		
	1	2	...	9	10	Min	Max	Avg
3	98.2	98.4	...	98.2	98	98	98.4	98.2
5	98.4	98	...	98	98.2	97.8	98.4	98.12
7	98.2	98	...	97.6	97.8	97.6	98.2	97.86
9	97.6	97.6	...	97.4	97.4	97.4	97.6	97.52
11	97.6	97.4	...	97.4	97.4	97.2	97.6	97.42
13	97	97.2	...	97.4	97.4	97	97.4	97.28
15	97.2	97.2	...	97	97	97	97.2	97.1
17	97.2	97.2	...	97.2	97	97	97.2	97.08
19	96.6	97	...	97	97	96.6	97.2	97
21	97.2	97	...	97	97.2	97	97.2	97.08
23	96.8	97	...	97.2	96.8	96.4	97.2	97
25	97.2	97.2	...	97	97	96.8	97.2	97

C. Pengujian dan Analisis

1) Nilai k Terbaik

Pengujian nilai *k* terbaik dilakukan dengan menggunakan tools *KNIME* dengan

menggunakan data training yang berjumlah 502 data, dimana terdapat 252 aktivitas tidak mencurigakan dan 250 aktivitas mencurigakan. Nilai *k* yang diuji yaitu dari $k = 3$ hingga $k = 25$. Nilai *k* terbaik ditentukan berdasarkan nilai akurasi paling tinggi dengan menggunakan *k-Fold Cross Validation*. *K-Fold Cross Validation* akan mengambil data secara random dari data training yang dimiliki, yang nantinya akan dijadikan sebagai data testing. Pada pengujian ini, data dibagi kedalam 10 partisi. Masing-masing dari 10 partisi, 9 partisi akan digunakan sebagai data training, dan 1 partisi akan digunakan sebagai data testing. Pengujian ini dilakukan untuk masing-masing nilai *k* sebanyak 10 kali. Berikut ini grafik hasil pengujian nilai *k* terbaik.



Gbr. 36 Grafik Akurasi Nilai k

Dari hasil pengujian yang dilakukan, nilai akurasi tertinggi yang didapatkan berada pada range 98% - 98,4% dengan rata-rata akurasi 98,2%. Terlihat bahwa nilai akurasi semakin menurun berdasarkan kenaikan nilai *k*. Berdasarkan nilai akurasi tersebut, maka penelitian ini menggunakan nilai $k = 3$. Beberapa kesalahan pendeteksian terjadi karena perubahan lingkungan dengan kondisi yang tidak signifikan. Setelah dilakukan kajian dan analisis, perubahan yang tidak signifikan didapatkan karena pengaruh bayangan manusia dari cahaya matahari yang masuk melalui jendela bertirai. Sehingga cahaya yang mempengaruhi agak sedikit samar.

2) Nilai Threshold

Pengujian ini dilakukan dengan menggunakan nilai *threshold* antara 5 – 60 dengan menggunakan kelipatan 5. Pengujian ini dilakukan dengan menggunakan 4 gambar yang berbeda untuk setiap label, yaitu aktivitas mencurigakan dan aktivitas tidak mencurigakan. Pengujian ini dilakukan untuk mendapatkan nilai *threshold* terbaik yang akan digunakan oleh system, agar system dapat melakukan proses ekstraksi ciri dengan baik tanpa terganggu noise yang ada pada citra. Berikut hasil pengujian *threshold*.

TABEL III. PENGUJIAN NILAI *TRESHOLD*

<i>Threshold</i>	Aktivitas							
	Tidak Mencurigakan				Mencurigakan			
5	-	-	-	-	-	-	-	-
10	-	-	-	-	-	-	-	-
15	-	-	-	-	-	-	-	-
20	-	-	-	-	-	-	-	-
25	-	-	-	-	-	-	-	-
30	-	-	-	-	-	-	-	-
35	✓	-	-	✓	✓	✓	✓	✓
40	-	-	-	-	✓	✓	✓	✓
45	✓	✓	✓	-	✓	✓	✓	✓
50	✓	✓	✓	✓	✓	✓	✓	✓
55	✓	✓	-	-	-	-	✓	-
60	✓	✓	-	✓	-	-	-	-

Setelah dilakukan pengujian nilai *threshold* dengan menggunakan 4 citra untuk masing-masing label, nilai *threshold* 45-50 dianggap dapat memisahkan objek dengan cukup baik. Oleh karena itu, berdasarkan pengujian yang dilakukan, nilai *threshold*.

IV. KESIMPULAN

Penelitian berhasil mengidentifikasi aktivitas mencurigakan dengan akurasi antara 98% - 98,4% dengan rata-rata akurasi 98,2%. Hasil ini didapatkan dengan menggunakan 502 data *training* yang diuji

dengan menggunakan *k-fold cross validation*. Sehingga data training dan metode yang diajukan dapat diimplementasikan untuk sistem pendeteksi aktivitas mencurigakan bagi peserta ujian masuk berbasis komputer.

Metode ini tentu memiliki kekurangan, salah satunya adalah perubahan lingkungan yang sangat signifikan, seperti pengaruh lampu dan lain-lain. Sehingga perlu dilakukan uji coba lanjutan untuk kondisi ruangan yang beragam agar metode dapat menyesuaikan diri dengan perubahan lingkungan yang tidak signifikan karena pengaruh cahaya matahari.

REFERENSI

- [1] Widyawan, Zul, M.I., & Nugroho, L.E. *Adaptive Motion Detection Algorithm using Frame Differences and Dynamic Template Matching Method*. The 9th International Conference on Ubiquitous Robots and Ambient Intelligence, 2–5, 2012
- [2] Zul, M.I., Muslim, I., & Hakim, L. *Human Activity Recognition by Using Nearest Neighbor Algorithm from Digital Image*. 2017 International Conference on Soft Computing, Intelligent System and Information Technology (ICSIT)
- [3] Zul, M.I., Widyawan & Nugroho, L.E.. *Deteksi Gerak dengan Menggunakan Metode Frame Differences pada IP Camera*. 5th International Conference on Information Technology and Electrical Engineering (CITEE). Yogyakarta: Jurusan Teknik Elektro dan Teknologi Informasi UGM 2012.
- [4] Kenchannavar, H. H., Patkar, G. S., Kulkarni, U. P., & dan Math, M. M. *Simulink Model for Frame Difference and Background Subtraction comparison in Visual Sensor Network*. The 3rd International Conference on Machine Vision (ICMV 2010). Hongkong.
- [5] Mishra, S., Mishra, P., Chaudhary, N. K., & dan Asthana, P. A Novel Comprehensive Method for Real Time Video Motion Detection Surveillance. International Journal of Scientific dan Engineering Research 2011.
- [6] J. Han, M. Kamber and J. Pei, *Data Mining Concepts and Techniques*, United State of America: Elsevier (The Morgan Kaufmann), 2011.