

Klasifikasi K-NN Dan Naive Bayes Terhadap Pelacakan Ujung Jari Berbasis Camera Smartphone

Sudarno¹, Agus Kristanto², Taman Ginting³

¹Teknik Elektro, Politeknik Prata Mulia Surakarta

^{2,3}Teknik Komputer. Politeknik Pratama Mulia Surakarta
isusay@gmail.com

ABSTRACT

Research vision and computer graphics, motion tracking fingertips of video image sequences automatically has been a very interesting study to be developed. Good tracking method should be able to find back the ends of the finger after the obstacle no longer exists and is also able to predict the fingertips are hindered by the position information of the fingertips that are not obstructed. In practice, there are two ways to track the movement of an object. The first approach is called the approach-by-detection tracking. In this approach, to track an object the object detection performed on each frame of moving pictures or videos are observed, in order to determine the position of the object in each frame. The second approach is referred to as a detection approach-by-tracking. From the results of tracking the movement of a hand camera-based smartphone will use a method of classification algorithm k-Nearest Neighborhood (k-NN) and Naive Bayes (NB). From the research results will be known influence on the movement of fingertips, and penangkapan fingertips as well as the classification method to obtain the arrest of the movement of a fingertip accuracy, algorithms and data from a tracking number will be tested for accuracy. The best results of testing of the algorithms will be the result and purpose of this research.

Keyword : Finger Detection, Gesture Recognition, Naif Bayes Algoritma

I. PENDAHULUAN

Dalam bidang penelitian visi dan grafika komputer, pelacakan gerak ujung-ujung jari dari sekuen citra video secara otomatis telah menjadi penelitian yang sangat menarik untuk dikembangkan. Hal ini disebabkan karena pelacakan gerak ujung-ujung jari merupakan komponen utama yang sangat diperlukan pada kebanyakan sistem pengenalan bahasa isyarat (*gesture recognition*), antarmuka alami (*natural user interface*), penangkap gerak (*motion capture*) untuk pembuatan animasi tiga-dimensi (3D), dan sistem aplikasi komputer untuk membantu operasi bedah di bidang medik dan kedokteran (*computer assisted surgery*).

Pelacakan ujung-ujung jari juga merupakan masalah yang sulit dan menantang. Tidak seperti

pelacakan gerak tangan, kaki, atau anggota tubuh yang lain seperti kepala, gerakan masing-masing ujung jari sangat bervariasi dan berubah secara cepat dengan luas daerah pengamatan atau ROI (*region of interest*) untuk masing-masing ujung jari relatif sangat kecil sehingga sangat sensitif terhadap derau yang disebabkan oleh citra latar belakang yang berubah-ubah atau sulit dibedakan dengan citra objek ujung-ujung jari yang hendak dilacak. Metode penjejakan yang baik harus segera dapat menemukan kembali ujung-ujung jari tersebut setelah halangan tersebut tidak ada lagi dan juga mampu memprediksi ujung-ujung jari yang terhalang berdasarkan informasi posisi dari ujung-ujung jari lain yang tidak terhalang.

Ada dua cara untuk melacak gerakan suatu objek. Pendekatan pertama disebut pendekatan *tracking-by-detection*. Pada pendekatan ini,

untuk melacak suatu objek dilakukan pendeteksian objek tersebut pada setiap frame gambar bergerak atau video yang diamati, guna menentukan posisi objek itu di setiap framanya. Pendekatan kedua disebut sebagai pendekatan *detection-by-tracking*. Segmentasi citra dalam prapengolahan dan analisis citra bertujuan untuk mempartisi gambar menjadi beberapa sub-daerah berdasarkan fitur yang diinginkan. Kontur aktif telah banyak digunakan sebagai metode segmentasi citra. Metode ini selalu menghasilkan sub-wilayah dengan batas-batas kontinyu, berbeda dengan metode tepi kernel berbasis deteksi, contohnya seperti detektor tepi Sobel, sering menghasilkan batas terputus-putus.

Metode Kontur aktif didasarkan pada penggunaan kontur yang dapat berubah bentuk (*deformable*) menyesuaikan diri dengan bentuk objek bergerak yang dijadikan target. Berdasarkan implementasi matematikanya, ada dua pendekatan utama dalam kontur aktif: *Snakes* dan *Level-set*. *Snakes* secara eksplisit menggeser titik-titik pada kontur yang telah ditetapkan menggunakan metode minimalisasi energi, sementara *level-set* mendekati kontur bergerak sebagai tingkat tertentu fungsi. Ada dua jenis model kontur aktif sebagai metode segmentasi citra yaitu: berdasarkan tepi dan berdasarkan wilayah.

Kontur aktif berbasis tepi menerapkan sebuah detektor tepi, biasanya didasarkan pada gradien gambar, untuk menemukan batas-batas sub-daerah dan untuk menggambarkan kontur ke batas terdeteksi. Kontur aktif berbasis tepi berhubungan erat dengan segmentasi berdasarkan tepi. Penggunaan teori *level-set* lebih fleksibel dan mudah diterapkan pada kontur aktif. Namun, kontur aktif berbasis wilayah lebih berlaku untuk gambar hanya relatif sederhana dimanakanilai piksel-piksel di dalam sub- daerah relatif seragam tanpa ada tepi internal .

II. TINJAUAN PUSTAKA

Smoothing atau juga disebut *Blur*, adalah operasi pengolahan gambar sederhana dan sering

digunakan. Hal ini biasanya dilakukan untuk mengurangi *noise* atau kamera artefak. Efek Gaussian Blur adalah filter yang memadukan sejumlah tertentu pixel secara bertahap, mengikuti kurva berbentuk lonceng (Bradski, 2008; Al-Amri, 2010)

Gaussian Blur dilakukan dengan *convolving* setiap titik dalam larik masukan dengan kernel Gaussian dan kemudian menjumlahkan untuk menghasilkan larik luaran. Berikut, fungsi Gaussian Blur (Bradski, 2008):

A. Deteksi Ujung Jari

Untuk menemukan ujung jari dapat dilakukan dengan dua langkah. Yaitu yang pertama menentukan kontur tangan. Yang kedua menemukan *convex* dan *defects*-nya. Convex hull merupakan cara untuk memberikan titik awal dan akhir dari suatu puncak atau kelengkungan dari telapak tangan. Setelah mendapatkan nilai atau poin-poin maka kita dapat menentukan nilai kecuramannya atau kedalaman antar lengkungan. *Defects* merupakan segmentasi yang terbentuk dari dua lengkungan yang telah dideteksi oleh *convex hull*.

B. Pelacakan Objek Berbasis Model

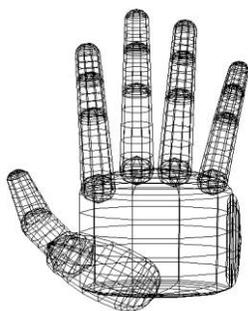
Pelacakan objek dilakukan dengan menganalisis dua model yaitu model fungsi dan model pengukuran. Model fungsi merupakan analisa pergerakan objek untuk tiap satuan waktu sedangkan model pengukuran merupakan analisa interaksi objek dengan lingkungannya (Arumpalam, 2002).

Analisa pergerakan objek dilakukan dengan pendekatan probabilitas. Dengan ini maka semua kemungkinan solusi pergerakan objek direpresentasikan dalam bentuk distribusi probabilitas. Kemudian dalam pencariannya, posisi objek dihitung berdasarkan derajat kepercayaan sehingga diperoleh posisi objek atau dengan kata lain objek terlacak. Objek yang akan dilacak memiliki *state* atau keadaan. *State* ini sendiri merupakan himpunan semua aspek atau fitur yang terdapat pada objek, misalkan posisi

untuk setiap satuan waktu pada sistem dinamis.

C. Model Tangan

Model tangan dibuat sebagai interpretasi pergerakan tangan dalam bentuk 3D. hal ini untuk membuat animasi gerakan tangan manusia yang dapat dilihat dari beberapa sudut pandang seperti bagian depan, belakang, samping, atas dan bawah yang tidak disediakan oleh media 2D. Pemodelan tangan kedalam 3D terdiri dari beberapa bagian tangan. Seperti yang telah di gambarkan oleh Stenger et al (2001) sebagai berikut.



Gambar 1. Model tangan

Pengerakan model tangan ini akan mengikuti pergerakan dari setiap titik engsel, dimana titik engsel merupakan titik hasil pelacakan pada tangan. Titik ini bergerak dalam ruang dimensi 3, sehingga pergerakan inilah yang membuat interaksianimasi pergerakan tangan antara komputer dan tangan menjadi hidup. Seperti yang telah dijelaskan dalam pelacakan objek, pergerakan titik merupakan hasil dari prediksi yang telah terkoreksi, yaitu dengan metode *particle filter*.

D. Penangkap Gerakan Jemari Tangan untuk Animasi 3D

Animasi adalah serangkaian gambar yang diberikan sehingga membentuk sebuah film. Kualitas film anda dikendalikan oleh semua fitur yang disebutkan di atas termasuk frame per detik (fps), ukuran output, jenis file dan kompresi (Cronister, 2009). Opsi dasar animasi adalah mengubah ukuran, rotasi dan lokasi

objek. Sedangkan animasi real-time memungkinkan menambahkan sifat fisik objek dan menggunakan keyboard dan fitur lain untuk mengendalikannya.

E. Metode K-Nearest Neighbour (K-NN)

Klasifikasi adalah proses untuk menemukan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang labelnya belum diketahui. Model itu sendiri dapat berupa aturan “jika-maka”, berupa *decision tree*, formula matematis atau *Neural Network*. Salah satu metode klasifikasi yaitu *Nearest Neighbor* (NN).

Metode NN adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Prinsip kerja NN adalah mencari jarak terdekat antara data yang akan dievaluasi dengan tetangga (*neighbor*) terdekatnya dalam data *training*. Dekat atau jauhnya tetangga biasanya dihitung berdasarkan jarak *euclidean*. Rumus jarak *euclidian* yang digunakan dalam metode NN untuk mengambil keputusan dengan menghitung jarak terpendek antara masukan dengan masing-masing data yang penentuan bobot atau labelnya dilakukan secara acak atau random yaitu:

$$d = \sqrt{\sum_{i=1}^p (x_q - X_q)^2} \dots\dots\dots (1)$$

d = Jarak euclidian

p = Jumlah jari

x = Data pada saat pengamatan atau data testing

X = Data uji

Pada kebanyakan kasus nilai $k=1$ dan $k=2$ menunjukkan hasil yang terbaik dalam tahap *positioning* (Otsason dkk, 2005).

F. Metode Naïve Bayes (NB)

Naïve Bayes adalah suatu probabilistik

simpel yang berdasarkan pada *teorema Bayes* pada umumnya, *inferensi Bayes* khususnya dengan asumsi independensi yang kuat (*naive*). Dalam melakukan klasifikasi data, *Naive Bayes* mengasumsikan bahwa ada atau tidak adanya suatu fitur pada suatu kelas tidak berhubungan dengan ada atau tidaknya fitur lain di kelas yang sama.

Umumnya kelompok atribut E direpresentasikan dengan sekumpulan nilai data. L adalah variable klasifikasi dalam kasus ini yaitu koordinat dan l adalah nilai dari L . Dari sudut pandang peluang berdasarkan aturan *bayes* ke dalam kelas l adalah:

$$P(l|E) = \frac{P(E|l)P(l)}{P(E)} \dots\dots\dots(2)$$

Untuk menentukan pilihan kelas, digunakan peluang maksimal dari seluruh l dalam L , dengan fungsi :

$$\arg \max_{l \in L} \frac{P(E|l)P(l)}{P(E)} \dots\dots\dots(3)$$

Karena nilai konstan untuk semua kelas, maka dapat diabaikan sehingga menghasilkan fungsi :

$$f_l(E) = \arg \max_{l \in L} P(E|l)P(l) \dots\dots\dots(4)$$

Mengatasi berbagai permasalahan, berbagai varian dari pengklasifikasian yang menggunakan *teorema Bayes* diajukan, salah satunya adalah NB :

$$f_l(E) = \arg \max_{l \in L} P(E|l)P(l) \prod_{j=1}^n P(E_j|l) \dots\dots\dots(5)$$

Pendekatan *Naive Bayes* untuk pemodelan kekuatan sinyal sebagai distribusi *Gaussian* dan menggunakan kekuatan sinyal yang dikumpulkan untuk mempelajari parameter distribusi *Gaussian*, yang merupakan *mean* dan standar deviasi dari data *training*. Seperti halnya menghitung jarak *euclidian* sinyal pengamatan vektor S di posisi l . S_i adalah penangkapan yang diamati dari jari di posisi l , M_i^l adalah rata-rata data penangkapan dari jari i di posisi l

dihitung dari data keseluruhan D_i^l adalah standar deviasi dari jari i di posisi l dihitung dari data keseluruhan penentuan *grid* dan $|P|$ adalah jumlah jari yang terbaca yang terbaca di posisi l . Ketika jari vektor S diperoleh dari pengukuran waktu saat ini maka probabilitas $P(S|l)$ dihitung untuk semua posisi dari setiap jari yang sudah menjadi data base. Posisi l yang memiliki probabilitas tertinggi $P(S|l)$ untuk vektor pendeteksian jari diklasifikasikan sebagai posisi pengguna di lapangan pada saat ini.

$$P(S|l) = \prod_{i=1}^{|P|} \left[\frac{1}{\sqrt{2\pi} (D_i^l)^2} \exp \left\{ -\frac{(S_i - M_i^l)^2}{2(D_i^l)^2} \right\} \right] \dots\dots\dots(6)$$

$$M_i^l = \frac{\sum_{i=1}^n X_i^l}{n} \dots\dots\dots(7)$$

$$D_i^l = \sqrt{\frac{\sum_{i=1}^n (X_i^l - M_i^l)^2}{n-1}} \dots\dots\dots(8)$$

dengan :

$P(S|l)$ = Probabilitas posterior jarak pengamatan terhadap posisi l

M_i^l = Rata-rata jarak dari jari i di posisi l

D_i^l = Standar deviasi dari jari D di posisi l

$|P|$ = Jumlah jari

III. METODE PENELITIAN

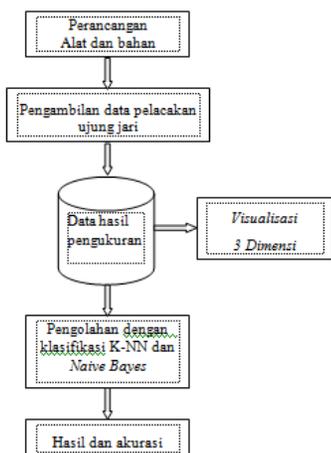
Tahapan penelitian dilakukan berdasarkan teori dan metode yang berhubungan dengan penelitian. Tahapan tersebut terkait dengan:

- (1) Perancangan peralatan pendukung penelitian terkait *smartphone* dan software 3 dimensi
- (2) Pengukuran nilai pergerakan jari yang

ditangkap oleh kamera dan lama waktu pengambilan untuk data.

- (3) Pengolahan data menggunakan metode klasifikasi K-NN dan Nive Bayes.
- (4) Pembuatan Visualisasi 3D
- (5) Penentuan hasil dan pembuatan laporan.

Diagram alir penelitian ini adalah sebagai berikut:



Gambar 2.. Metode Penelitian

A. Alat dan bahan.

Bahan penelitian yang digunakan klasifikasi k-nn dan naive bayes terhadap pelacakan ujung jari berbasis camera diperlukan perangkat keras dan perangkat lunak sebagai berikut:

1. Perangkat keras (*Hardware*)
 - a. 1 buah Smartphon
 - b. Laptop (*Windows*)
 - c. Kabel Data
Kabel data digunakan untuk menghubungkan semarphon dengan laptop.
2. Perangkat Lunak
 - a) *Blender 3D* perangkat lunak yang dapat digunakan untuk membuat animasi.
 - b) *Rapidminer 5.1.012*
Rapidminer berfungsi sebagai *tool* untuk aplikasi *Nearest Neighbor (NN)*. dan naive bayes *RapidMiner* juga berfungsi untuk memvisualisasikan data

pengukuran format data yang digunakan adalah format *csv ().

- c) *Ms Excel 2007*
Digunakan dalam pengolahan data dan pembuatan visualisasi grafik.
- d) *Photoshop*
Digunakan dalam pembuatan gambar desain jari 2D.

3. Pengujian *Pre-Processing*

Pengujian berikut adalah pengujian data citra pada hasil pengolahan citra, dengan melihat tampilan data citra yang sudah dikonversikan ke ruang warna HSV. Kemudian dilakukan *color filtering* untuk mendeteksi warna kulit tangan menggunakan *thresholding*.

Untuk tahap proses *pre-processing* diperlukan konversi ruang warna RGB ke HSV, berikut adalah hasil konversi :



Gambar 3. Hasil konversi RGB ke HSV

IV. PEMBAHASAN

A. Pengujian Metode *Template Matching*

Pada tabel berikut adalah lima kelompok citra sampel yang dijadikan *template* untuk pembandingan dengan citra-citra *streaming* pada pengujian :

Tabel 1. kelompok citra sampel

	Sampel Value1	Sampel Value2	Sampel Value3	Sampel Value4	Sampel Value5
a					
b					
c					

Berikut seperti pada tabel2 sampai pada

tabel7 adalah hasil perbandingan beberapa citra sampel (*template*) value1 sampai dengan value5 dengan beberapa citra *streaming* :

Tabel 2. Hasil pengujian citra value1-5 dengan model beberapa citra *streaming*

Sampel Value	Coba1	Coba2	Coba3	Coba4
1a	0,114113	0,454561	0,795432	0,790518
1b	0,301332	0,513142	0,571363	0,751228
1c	0,345635	0,888943	0,733373	0,740587
2a	0,734525	0,832323	0,785233	0,914378
2b	0,765522	0,852228	0,801255	0,924342
2c	0,778266	0,812441	0,745129	0,915674
3a	0,867433	0,910078	0,672771	0,882079
3b	0,861343	0,887433	0,767432	0,828006
3c	0,852249	0,838722	0,719643	0,876597
4a	0,894005	0,924566	0,891233	0,701926
4b	0,914659	0,894821	0,834819	0,684608
4c	0,894034	0,927908	0,883441	0,670085
5a	0,984732	0,934842	0,923331	0,672308
5b	0,978822	0,959374	0,954652	0,672907
5c	0,979331	0,923731	0,984732	0,628474

Sampel Value	Coba5	Coba6	Coba7	Coba8
1a	0,623321	0,562308	0,664408	0,770518
1b	0,650045	0,537249	0,620786	0,751228
1c	0,578943	0,512354	0,664823	0,740587
2a	0,065443	0,393249	0,774565	0,914378
2b	0,241327	0,513142	0,417364	0,924342
2b	0,319835	0,378972	0,425885	0,915674
3a	0,583109	0,633247	0,628831	0,882079
3b	0,556261	0,618946	0,601287	0,828006
3c	0,574012	0,629801	0,640719	0,876597
4a	0,643551	0,654012	0,644012	0,701926
4b	0,616557	0,629644	0,624231	0,684608
4c	0,612387	0,607890	0,610089	0,670085
5a	0,845341	0,895646	0,886352	0,672308
5b	0,862989	0,890065	0,824971	0,672907
5c	0,979331	0,923731	0,984732	0,628474

Sampel Value	Coba9	Coba10	Coba11	Coba12
1a	0,867433	0,823321	0,872301	0,770518
1b	0,867433	0,826466	0,851535	0,751228
1c	0,860089	0,823321	0,842371	0,740587
2a	0,745341	0,746317	0,741224	0,914378
2b	0,723367	0,66164	0,713982	0,924342
2c	0,645341	0,706631	0,631004	0,915674
3a	0,063386	0,221864	0,435615	0,882079
3b	0,232126	0,453142	0,401237	0,828006
3c	0,369831	0,392972	0,442885	0,876597
4a	0,572412	0,552391	0,620004	0,671926
4b	0,584155	0,552391	0,645904	0,684608
4c	0,572841	0,534539	0,681344	0,670085
5a	0,610256	0,630056	0,657293	0,692308
5b	0,685673	0,630056	0,654338	0,672907
5c	0,624675	0,672361	0,657271	0,628474

Sampel Value	Coba13	Coba14	Coba15	Coba16
1a	0,894005	0,895630	0,898305	0,770518
1b	0,921469	0,917759	0,924239	0,751228
1c	0,856814	0,824034	0,871134	0,740587
2a	0,643411	0,643551	0,743551	0,914378
2b	0,616557	0,610057	0,716557	0,924342
2c	0,666087	0,685201	0,712727	0,915674
3a	0,572455	0,552319	0,620094	0,882079
3b	0,584155	0,552691	0,642904	0,828006
3c	0,572418	0,534039	0,681347	0,876597
4a	0,023311	0,151864	0,412281	0,651926
4b	0,312126	0,308112	0,371293	0,644608
4c	0,209422	0,370032	0,473449	0,670085
5a	0,512329	0,508882	0,512934	0,692308
5b	0,542938	0,519002	0,620001	0,672907
5c	0,523221	0,551346	0,629381	0,628474

Sampel Value	Coba17	Coba18	Coba19	Coba20
1a	0,988652	0,984732	0,912732	0,790518
1b	0,975422	0,961622	0,934822	0,751228
1c	0,970011	0,976131	0,953331	0,740587
2a	0,845341	0,834187	0,845354	0,914378
2b	0,862989	0,862211	0,862007	0,924342
2c	0,879331	0,979331	0,879767	0,915674
3a	0,685273	0,730056	0,754342	0,882079
3b	0,685653	0,710056	0,754312	0,828006
3c	0,785673	0,730767	0,696338	0,876597
4a	0,545673	0,520056	0,614338	0,701926
4b	0,519239	0,4951882	0,528029	0,684608
4c	0,519912	0,519002	0,513391	0,670085
5a	0,067999	0,218647	0,478366	0,672308
5b	0,172126	0,208112	0,472123	0,672907
5c	0,289193	0,270992	0,493418	0,628474

Dengan melihat hasil pada tabel di atas, dapat disimpulkan bahwa pengujian *template matching* sudah berjalan dengan baik. Dari citra sampel (*template*) yang telah tersimpan, kemudian dibandingkan dengan citra *streaming*, maka akan didapatkan nilai masing-masing *threshold* rata-rata dari setiap citra *streaming*. Nilai yang berwarna merah adalah warna yang ikut diproses karna paling mendekati kemiripan. Kemudian *streaming* yang memiliki nilai rata-rata terkecil ,maka itulah citra yang paling mirip dengan citra sampel (*template*).

B. Pengujian Sistem Keseluruhan

Pengujian Pengujian terakhir adalah pengujian sistem secara keseluruhan dari awal hingga akhir, dimana pengujian dilakukan dengan menjalankan aplikasi secara

keseluruhan. Memberikan input isyarat tangan pada kamera PC yang diambil dari jarak kurang lebih 100 cm sampai dengan 150 cm, maka Isyarat masih dapat diterima oleh kamera. Eksperimen yang akan dilakukan adalah melihat kinerja dari dua Algoritma klasifikasi dokumen teks yaitu algoritma Naive Bayes dan algoritma k-Nearest Neighbor

C. Pengujian Data metode K-NN

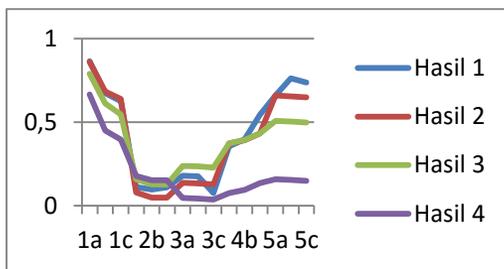
Metode k-Nearest Neighbor merupakan salah satu metode berbasis NN yang paling populer. Nilai k yang digunakan menyatakan jumlah tetangga terdekat yang dilibatkan dalam penentuan prediksi label kelas pada data uji. Untuk memperkirakan nilai k yang terbaik, bisa dilakukan dengan menggunakan teknik validasi silang (Cross Validation). Jika nilai k terlalu kecil, maka berakibat hasil prediksi yang didapat bisa sensitif terhadap keberadaan noise, namun jika k terlalu besar maka tetangga terdekat yang terpilih mungkin terlalu banyak dari kelas lain yang sebenarnya tidak relevan karena jarak yang terlalu jauh. Berikut adalah hasil pengujian data dengan cara melakukan uji coba memasukan nilai k (jumlah tetangga terdekat)

Tabel 3 Data Ukur

coba 1	coba 2	coba 3	coba 4	Pengukuran
0.124123	0.454561	0.795432	0.790518	1a
0.301332	0.513142	0.571363	0.751228	1b
0.345635	0.888943	0.734511	0.915723	1c
0.734511	0.852225	0.785512	0.955721	2a
0.765512	0.832455	0.805112	0.980543	2b
0.765112	0.812441	0.790518	0.920011	2c
0.867432	0.912002	0.751228	0.910019	3a
0.864433	0.887213	0.715723	0.882012	3b
0.857432	0.832144	0.888943	0.872345	3c
0.895721	0.903211	0.832225	0.702134	4a
0.915723	0.890567	0.832455	0.681234	4b
0.955721	0.923451	0.882441	0.673432	4c
0.980543	0.932145	0.980543	0.621852	5a
0.976623	0.954213	0.920011	0.675432	5b
0.969332	0.923433	0.910019	0.621852	5c

Tabel 4. Data Uji

coba 1	coba 2	coba 3	coba 4	Pengukuran
0.988652	0.984732	0.912731	0.790518	1a
0.975412	0.984732	0.912731	0.751228	1b
0.970001	0.984732	0.891232	0.740587	1c
0.845534	0.812345	0.891232	0.911111	2a
0.861234	0.812345	0.891232	0.916574	2b
0.874321	0.812345	0.891232	0.916574	2c
0.689321	0.731241	0.630123	0.822131	3a
0.689742	0.731241	0.630123	0.822131	3b
0.783214	0.731241	0.630123	0.822131	3c
0.541231	0.523413	0.523188	0.822131	4a
0.519239	0.523413	0.523188	0.822131	4b
0.412315	0.523413	0.523188	0.822131	4c
0.321451	0.321456	0.472312	0.822131	5a
0.213451	0.321456	0.472312	0.822131	5b
0.231142	0.321456	0.472312	0.822131	5c

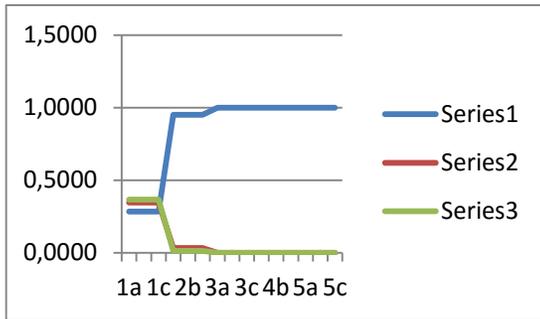


Gambar 4. Grafik Hasil Pengolahan menggunakan K-NN

Dari data diatas maka data percobaan rata rata adalah

Tabel 6. Rata – rata Pengukuran dari percobaan

percobaan	a	b	c
1a	0.2856	0.3465	0.3678
1b	0.2856	0.3465	0.3678
1c	0.2856	0.3465	0.3678
2a	0.9514	0.0347	0.0139
2b	0.9514	0.0347	0.0139
2c	0.9514	0.0347	0.0139
3a	1.0000	0.0000	0.0000
3b	1.0000	0.0000	0.0000
3c	1.0000	0.0000	0.0000
4a	1.0000	0.0000	0.0000
4b	1.0000	0.0000	0.0000
4c	1.0000	0.0000	0.0000
5a	1.0000	0.0000	0.0000
5b	1.0000	0.0000	0.0000
5c	1.0000	0.0000	0.0000



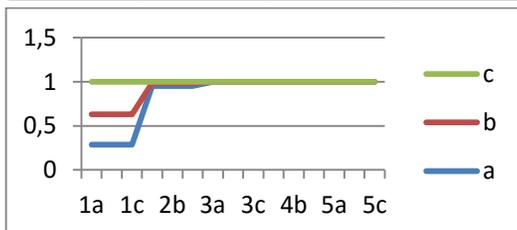
Gambar 5. Grafik Rata – Rata Pengukuran

D. Implementasi Algoritma Naïve Bayes

Implementasi pengujian menggunakan naïve bayes merupakan hasil implementasi yang lebih baik dari K-NN. Terlihat dari accuracy besaran nilai yang dihasilkan.

Tabel 7. Nilai Akurasi menggunakan Naïve Bayes

percobaan	a	b	c
1a	0.285632	0.346532	0.367837
1b	0.285632	0.346532	0.367837
1c	0.285632	0.346532	0.367837
2a	0.951392	0.034664	0.013944
2b	0.951392	0.034664	0.013944
2c	0.951392	0.034664	0.013944
3a	0.999994	5.61E-06	1.71E-07
3b	0.999994	5.61E-06	1.71E-07
3c	0.999994	5.61E-06	1.71E-07
4a	1	4.42E-12	2.05E-15
4b	1	4.42E-12	2.05E-15
4c	1	4.42E-12	2.05E-15
5a	1	1.79E-20	2.51E-26
5b	1	1.79E-20	2.51E-26
5c	1	1.79E-20	2.51E-26



Gambar 6. Grafik Nilai Menggunakan Naïve Bayes

V. KESIMPULAN

Dari hasil penelitian yang telah dibahas maka dihasilkan kesimpulan bahwa setelah algoritma Naïve Bayes dan algoritma k-Nearest Neighbor diimplementasikan dalam pendekteksian ujung jari menggunakan kamera maka pengklasifikasian dokumen sangat perlu untuk melakuakn perhitungan akurasi antara kedua algoritma. Dari kedua algoritma tersebut kinerja terbaik diperoleh setelah melakukan pengolahan data terlebih dahulu. Hasil terbaik ditunjukkan oleh algoritma naïve bayes.

REFERENSI

Alecu, F. (2010). *Blender Institute – the Institute for Open 3D Projects*. Open Source Science Journal. Vol. 2, No. 1, 2010.

Arulampalam, M. Sanjeev., Maskell S., Gordon N and Clapp T. (2002). *A tutorial on particle Filter for Online Nonlinier/Non Gaussian Bayesian Tracking*. IEEE Transaction on Signal Processing. Vol 40 No 2.

Bimbo, D and Dini, F. (2011). *Particle Filter-Based Visual Tracking With a First Order DynamicModel and Uncertainty Adaptation*. Computer Vision and Image Understanding, Vol. 115, No. 6. Hal(771–786).

Bradski, G. & Adrian. (2008). *Kehler Learning OpenCV Computer vision with OpenCV Library*. O, 'Reilly

B. Stenger, P. Mendonca, and R. Cipolla. (2001). *Model-based 3D tracking of an articulated hand*. CVPR.

D. Serby, E. K. Meier & L. V. Gool. (2004). *Probabilistic Object Tracking Using Multiple Features*. Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04). Hal(43-53).

- Feng et al. (2012). *Initialization of 3D Human Hand Model and Its Applications in Human*
- Hand Tracking. Journal Of Computers. Vol. 7, No. 2. Hal(419-426)
- Hyndman, R.J., dan A.B. Koehler. (2006). *Another look at measures of forecast accuracy.*
- International Journal of Forecasting 22, no. 4. Hal(679-688).
- Pachoulakis, I & Kapetanakis, K. *Augmented Reality Platforms For Virtual Fitting Rooms.* The International Journal of Multimedia & Its Applications (IJMA). Vol.4, No.4, August 2012
- P. Kumar, H. Weimin, I. U. Gu & Q. Tian. (2004). *Statistical modeling of complex backgrounds for foreground object detection.* IEEE Transaction on Image Processing. Vol 13, No. 1. Hal(43-53).
- Siradjuddin, Indah A. (2007). *Object Tracking in Image Sequence Using Particle Filter.*
- International Conference on Soft Computing, *Intelegent System and Information of Technology (ICSIT)*. Hal (260-265).
- Skala, V. (2012). *Holography, Stereoscopy and Blender 3D.* Proceeding Recent Research in Communications and Computers. Hal (172-176).
- Warade, S. et al. (2012). *Automated Training And Maintenance Through Kinect.*
- International Journal of Computer Science, Engineering and Applications (IJCSEA). Vol.2, No.3.