Pembangunan Aplikasi Deteksi dan Tracking Warna *Virtual Drawing* Menggunakan Algoritma *Color Filtering*

Rio Pradhitya

Teknik Informatika – Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
Email : riopradhitya@gmail.com

ABSTRAK

Berbagai ilmu dan teknik pengolahan citra yang ada pada saat ini sangat bervariasi. Penelitian dan pengembangan telah dilakukan ke arah deteksi dan *tracking* objek. Warna merupakan salah satu parameter yang digunakan deteksi dan *tracking* objek. Manusia dapat membedakan sebuah warna, tetapi komputer belum tentu dapat mengenali warna tersebut. Teknik pengolahan citra yang dapat mengenali warna, salah satunya adalah *color filtering*. *Color filtering* merupakan teknik pengolahan citra berdasarkan warna yang spesifik. Dalam penelitian ini, pembangun aplikasi deteksi dan *tracking* warna *virtual drawing* memanfaatkan kamera web (*webcam*) dan warna objek sebagai *marker*.

Warna objek yang digunakan pada aplikasi ini adalah merah. Pengolahan citra digunakan untuk membandingkan warna komponen dengan warna spesifik. Warna komponen diambil dari nilai *range* RGB (*Red*, *Green*, *Blue*). Sistem dibangun dengan menggunakan *library* tambahan yaitu AForge.NET *Framework*. Tujuan dari aplikasi ini dapat digunakan untuk menggambar secara virtual tanpa menggunakan mouse sebagai alat bantu pembelajaran seperti keperluan dalam presentasi agar lebih menarik.

Pengujian aplikasi menggunakan perangkat tambahan kamera 3MP dengan kondisi ruangan menggunakan lampu pijar 9 watt dan cahaya matahari. Berdasarkan hasil pengujian *black box* dan kuesioner, kesimpulan yang didapatkan bahwa aplikasi ini dapat diterima dalam mendeteksi dan *tracking* objek dan aplikasi ini dapat digunakan untuk menggambar secara *virtual* tanpa menggunakan *mouse.*

Kata Kunci : Deteksi, *Tracking*, Objek, *Color Filtering*

1. PENDAHULUAN

Pengolahan citra digital adalah pemrosesan citra, khususnya dengan menggunakan komputer sehingga citra itu kualitasnya menjadi lebih baik dan menghasilkan informasi nilai untuk tiap-tiap warnanya [1]. Hubungan pengolahan citra dengan *computer vision* yaitu pengolahan citra merupakan proses awal (*preprocessing*) pada *computer vision* [2].

Computer vision adalah transformasi atau perubahan dari data-data yang dapat berupa gambar diam ataupun *video* kamera menjadi bentuk lain atau satu representasi baru. *Computer vision* dapat didefinisikan dengan pengertian pengolahan citra yang dikaitkan dengan akuisisi citra, pemrosesan, klasifikasi, penganan, dan pencakupan keseluruhan, pengambilan keputusan yang diikuti pengidentifikasian citra [3].

Salah satu perkembangan dari penggunaan teknik *computer vision* adalah *tracking* objek. *Tracking* objek bertujuan untuk mendeteksi dan mengikuti posisi dari suatu objek bergerak yang diinginkan. *Tracking* objek banyak dibutuhkan oleh berbagai macam aplikasi *vision based* seperti *human computer interface*, kompresi/komunikasi *video* dan sistem keamanan [4]. *Tracking* objek harus mampu mendeteksi objek yang bergerak, memfilter *noise*, dan gerakan-gerakan lain yang tidak diperlukan.

Terdapat penelitian yang pernah dilakukan sebelumnya berkaitan dengan *tracking* objek berbasis warna, diantaranya penelitian yang dilakukan oleh Umar dengan judul penelitian "*Tracking* Arah Gerakan Telunjuk Jari Berbasis *Webcam* Menggunakan Metode Optical Flow" [5]. Pada hasil penelitian dapat disimpulkan, deteksi gerakan yang dilakukan pada aplikasi tersebut berupa gerakan sekuen yang berpindah dari posisi awal dan akhir. Aplikasi ini mempunyai manfaat yang baik untuk perkembangan teknologi selanjutnya, hanya saja pada kesimpulan penelitian aplikasi ini tidak dilanjutkan ke arah pendeteksian dan *tracking* objek secara *real-time*.

Penelitian selanjutnya dilakukan oleh Gentang, dalam penelitian yang berjudul "*Tracking* Bola Menggunakan Robotino". Pada penelitian yang telah dilakukan, metode yang dilakukan untuk *tracking* objek yaitu menggunakan metode *color filtering*. Pada hasil penelitian metode ini sudah berhasil melakukan deteksi dan *tracking* secara *real-time*, hanya saja keberhasilannya tidak terlalu besar.

Faktor yang menjadi kendala bukan dari metodenya tetapi implementasi terhadap robotnya.

Color filtering adalah suatu teknik pengolahan citra berdasarkan warna yang spesifik. Cara kerja pada *color filtering* adalah dengan membandingkan komponen warna setiap piksel citra dengan warna spesifik. *Color filtering* ini digunakan pada tahapan awal untuk mendeteksi objek berbasis warna, setelah menemukan warna yang telah ditentukan, penelitian dapat dilanjutkan ke proses *tracking* objek. Kelebihan menggunakan metode *color filtering* yaitu salah satu metode yang paling sederhana melakukan penyaringan warna dalam ruang warna RGB (*Red*, *Green*, *Blue*) yang memberikan performa tercepat pada gambar RGB [6].

Berdasarkan hal yang telah dijelaskan maka metode *color filtering* ini akan digunakan dalam penelitian untuk mengembangkan penelitian Umar dengan penelitian Gentang yang menjadikan aplikasi deteksi dan *tracking* objek berbasis warna dapat dilakukan secara *real-time*. Aplikasi yang akan dibangun pada penelitian ini akan dinamakan *virtual drawing*.

2. LANDASAN TEORI

2.1 Citra Digital

Citra diartikan secara harfiah adalah gambar pada bidang dwimatra (dua dimensi) yang disusun oleh banyak piksel yaitu bagian terkecil dari citra [2]. Dalam komputer, setiap piksel diwakili oleh dua buah bilangan bulat (*integer*) untuk menunjukkan lokasinya dalam bidang citra dan sebuah nilai dalam bilangan bulat (*integer*) untuk menunjukkan cahaya atau keadaan terang-gelap piksel tersebut.

Suatu piksel dapat ditunjukkan lokasinya dengan menggunakan koordinat (0,0) yang digunakan untuk posisi kiri atas dalam bidang citra. Tingkat pencahayaan suatu piksel dapat ditunjukkan dengan bilangan bulat yang besarnya 8 bit, dengan lebar selang nilai 0-255, di mana 0 untuk warna hitam, 255 untuk warna putih [1]. Citra digital yang diperoleh dari kamera digital akan menghasilkan intensitas pantulan yang menggambarkan terang dan gelap pada penampilan piksel-piksel penyusunannya selain itu juga akan memberikan informasi warna.

2.2 Pengolahan Citra

Pengolahan citra digital adalah pemrosesan citra, khususnya dengan menggunakan komputer sehingga citra itu kualitasnya menjadi lebih baik dan menghasilkan informasi nilai untuk tiap-tiap warnanya [1]. Tujuan utama pengolahan citra adalah agar citra yang mengalami gangguan mudah diinterpretasi oleh manusia maupun mesin (komputer).

Teknik pengolahan citra digital adalah mentransformasikan citra dua dimensi menjadi citra lain dengan kualitas yang lebih baik menggunakan komputer [2]. Proses ini mempunyai ciri data masukan dan informasi keluaran yang berbentuk citra. Jadi masukannya berupa citra dan keluarannya juga berbentuk citra, dengan kualitas yang lebih baik dari citra masukan.

Beberapa contoh operasi pengolahan citra adalah pengubahan kontras citra, penghilangan derau (*noise*) dengan operasi penapisan (*filtering*), penajaman (sharpening), pemberian warna semu (pseudo*color*ing), dan sebagainya.

2.3 *Computer Vision*

Pengolahan citra adalah pengolahan suatu citra dengan menggunakan komputer, untuk menghasilkan suatu citra yang lain. Hubungan pengolahan citra dengan *computer vision* yaitu pengolahan citra merupakan proses awal (*preprocessing*) pada *computer Vision* [2]. *Computer vision* mencoba meniru cara kerja visual manusia (*Human Vision*). *Human Vision* sesungguhnya sangat kompleks, manusia melihat objek dengan indera penglihatan (mata) lalu objek citra diteruskan ke otak untuk diinterpretasi sehingga manusia mengerti objek apa yang tampak dalam pandangan matanya.

Computer vision merupakan proses otomatis yang mengintegrasikan sejumlah besar proses untuk persepsi visual, seperti akuisisi citra, pengolahan citra, klasifikasi, pengenalan (*recognition*), dan membuat keputusan. *Computer vision* terdiri dari teknik-teknik untuk mengestimasi ciri-ciri objek dalam citra, pengukuran ciri yang berkaitan dengan geometri objek, dan menginterpretasi informasi geometri tersebut.

Proses-proses dalam *computer vision* dibagi dalam 3 (tiga) aktifitas :

1. Memperoleh atau mengakuisisi citra digital.
2. Melakukan teknik komputasi untuk memproses atau memodifikasi data citra. (operasi-operasi pengolahan citra).
3. Menganalisis dan menginterpretasi Citra menggunakan hasil pemrosesan untuk tujuan tertentu, misalnya memandu robot, mengontrol peralatan, memantau manufaktur dan lain-lain.

Tujuan dari *computer vision* adalah untuk mengekstrak informasi yang berguna dari gambar [7].

2.4 Konsep Citra *Gray* dan Warna

Warna adalah kesan yang diperoleh mata dari cahaya yang dipantulkan oleh benda-benda yang dikenalinya. Warna yang diterima oleh mata dari sebuah objek ditentukan oleh warna sinar yang dipantulkan oleh objek tersebut. Warna-warna yang dapat ditangkap oleh mata manusia merupakan kombinasi cahaya dengan panjang berbeda.

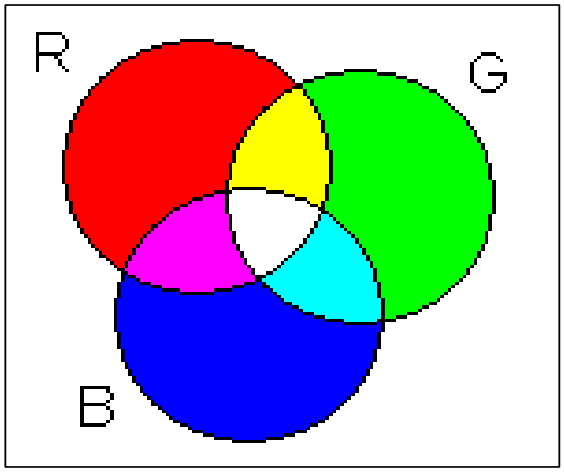
Citra *Gray* merupakan warna-warna piksel yang berada pada rentang warna hitam dan putih. Jumlah warna pada citra *gray* adalah 256, karena citra *gray*

jumlah bitnya adalah 8, sehingga jumlah warnanya adalah 28 = 256, nilanya berada pada jangkauan 0-255 [8] . Sehingga nilai intensitas dari citra *gray* tidak akan melebihi 255 dan tidak memungkinkan kurang dari 0.



Gambar 1 Contoh Representasi *Grayscale*

Sedangkan jumlah warna untuk citra RGB (*Red, Green Blue*) adalah dengan mengalikan jumlah pada masing-masing komponennya, jumlah dari tiap komponennya, R=255 (8 bit) G=255(8 bit) dan B=255(8 bit) sehingga sering kali disebut citra dengan intensitas 24 bit.



Gambar 2 Contoh Representasi RGB

2.5 Pengolahan Warna RGB

Pengolahan warna menggunakan model RGB sangat mudah dan sederhana, karena informasi warna dalam komputer sudah dikemas dalam model yang sama [1]. Hal yang perlu dilakukan adalah melakukan pembacaan nilai-nilai R,G, dan B pada suatu piksel.

Salah satu cara untuk menghitung warna yaitu dengan cara normalisasi ketiga komponen tersebut. Normalisasi penting dilakukan terutama bila sejumlah citra ditangkap dengan penerangan yang berbeda-beda. Hasil perhitungan tiap komponen warna pokok yang telah dinormalisasi akan menghilangkan pengaruh penerangan sehingga setiap komponen warna dapat dibandingkan satu sama lainnya walaupun berasal dari citra dengan kondisi penerangan yang tidak sama.

Cara melakukan normalisasi adalah sebagai berikut [1] :

$$r = \frac{R}{R+G+B}$$

$$g = \frac{G}{R+G+B}$$

$$b = \frac{B}{R+G+B}$$

Bila ketiga komponen warna yang telah dinormalisasi dengan masing-masing indeks warna merah (r), indeks warna hijau (g), dan indeks berwarna biru (b), mempunyai nilai yang sama (1/3), maka objek tidak berwarna. Bila r lebih besar dari pada g dan b, maka objek berwarna merah, dan seterusnya.

2.6 Format RGB (*Red,Green,Blue*)

Format RGB adalah format dasar yang digunakan oleh banyak peralatan elektronik seperti monitor, LCD atau TV untuk menampilkan sebuah gambar [9]. Pada format RGB, suatu warna didefinisikan sebagai kombinasi (campuran) dari komponen *Red*, *Green*, dan *Blue*. Nilai-nilai yang terdapat pada format RGB 24 bit yang dijelaskan pada pada tabel 1

Tabel 1 Format RGB

Warna	R	G	B
Hitam	0	0	0
Merah	255	0	0
Hijau	0	255	0
Biru	0	0	255
Kuning	255	255	0
Magenta	255	0	255
Cyan	0	255	255
Putih	255	255	255
Abu-Abu	127	127	127
Orange	255	110	0
Ungu	128	0	255
Coklat	128	25	0
Pink	255	190	220
Navy	0	0	120

2.7 *Color Filtering*

Color filtering adalah satu teknik pengolahan citra yang berguna untuk menemukan sebuah warna yang terdapat pada sebuah gambar [10]. Algoritma ini banyak digunakan dalam penelitian. Salah satunya pada *AForge.NET Framework* [11]. Pada

dasarnya pencarian ini menggunakan kombinasi dari komponen *Red*, *Green*, dan *Blue*.

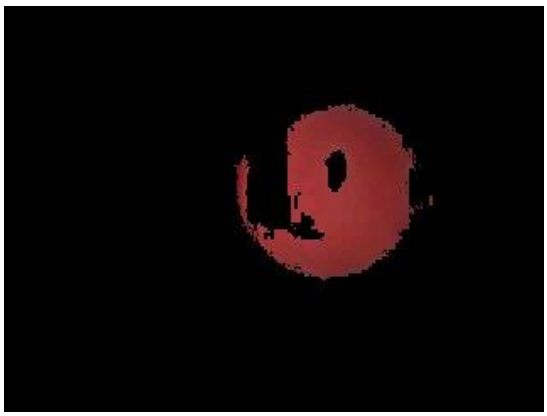
Cara kerjanya adalah dengan membandingkan komponen warna setiap piksel citra dengan warna spesifik. Apabila warnanya sesuai dengan warna spesifik komponen warna piksel tersebut maka dibiarkan saja. Namun, bila warnanya tidak sesuai dengan warna spesifik maka komponen warna piksel tersebut diubah menjadi warna *background*, biasanya menjadi warna hitam.

Warna yang digunakan dalam *Color Filtering* dapat direpresentasikan dalam berbagai ruang warna. Ada beberapa ruang warna yang dikenal, antara lain RGB (*Red*, *Green*, *Blue*), HSL (*Hue*, *Saturation*, *Luminance*), YCbCr, dan sebagainya. Dalam penelitian ini, peneliti menggunakan ruang warna RGB.

Terdapat penelitian yang dilakukan oleh Andrew Krillov yang menggunakan metode *color filtering* pada warna bola berwarna merah. Contoh hasil penggunaan *color filtering* [11]:



Gambar 3 Gambar Awal Sebelum Penggunaan *Color Filtering*



Gambar 4 Gambar Akhir Setelah Penggunaan *Color Filtering*

2.8 *Blobs Processing*

Blobs processing adalah suatu teknik pengolahan citra yang digunakan untuk mengenali suatu objek secara utuh [11]. Berikut beberapa penggunaan *blobs processing*:

2.8.1 *Blob Counter*

Blob counter digunakan untuk mencari objek, menghitung objek, mengekstrak objek dan mendapatkan dimensi objek yang ada pada gambar, yang dipisahkan dari latar belakang. Sebagai contoh penggunaan *Blob counter* terdapat pada penjelasan di bawah ini:



Gambar 5 Gambar Sumber

Dengan menggunakan algoritma *blob counter* dapat menentukan dan mencari objek berdasarkan ukuran terbesar dan mengekstrak gambar sebagai *blob* terbesar. Maka hasil dari algoritma *blob counter* tersebut, seperti gambar berikut:



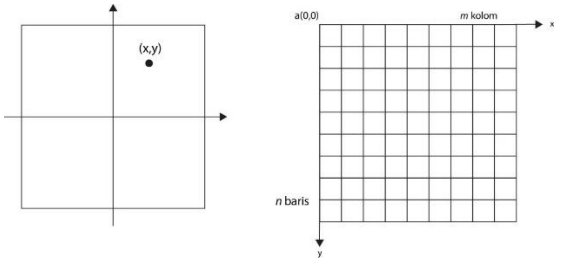
Gambar 6 Gambar hasil proses *Blob Counter*

Penggunaan *blob processing* sangat berguna bagi aplikasi *virtual drawing* yang akan dibangun, karena penentuan objek berbasis warna sangat berpengaruh pada penelitian ini.

2.9 Piksel

Sebuah piksel adalah sampel dari pemandangan yang mengandung intensitas citra yang dinyatakan dalam bilangan bulat [1]. Sebuah citra adalah kumpulan piksel-piksel yang disusun dalam larik

dua dimensi. Indeks baris dan kolom (x,y) dari sebuah piksel yang dinyatakan dalam bilangan bulat. Piksel (0,0) terletak pada sudut kiri atas pada citra, indeks bergerak ke kanan dan indeks bergerak ke bawah.



Gambar 7 Hubungan antara koordinat pada citra dan indeks larik

2.10 *Bitmap*

Bitmap adalah representasi dari citra grafis, seperti ukuran, format piksel yang merepresentasikan warna disimpan pada lokasi tertentu [2]. Saat ditampilkan pada layar komputer, setiap elemen digambarkan menjadi sebuah piksel tunggal.

2.11 Kamera *Web (Webcam)*

Kamera *Web* atau *Web Camera* disingkat *Webcam* adalah kamera *video* yang terhubung dengan komputer yang diletakkan ditempat tertentu dan tampilannya dapat dilihat pada sebuah jaringan (internet) [12]. Sesuai dengan sebutannya *Webcam* digunakan untuk aktivitas sebuah jaringan seperti komunikasi atau *broadcasting* secara langsung di internet. *Webcam* lebih sederhana jika dibandingkan dengan kamera *video* lainnya.

Keuntungan dari *Webcam* harganya murah dan mudah dalam menggunakannya. Perkembangan teknologi dan minat penggunaan komunikasi secara *online* membuat tampilan komputer jadi lebih terintegrasi dengan kamera (*built-in*) seperti laptop.

2.12 Aforge.NET

AForge.NET merupakan *open source Framework* yang dibangun oleh Andrew Kirilliov. *AForge.NET* merupakan *C# Framework* yang sangat membantu para developers dan penelitian khususnya di bidang *computer vision* dan kecerdasan buatan – pengolahan citra, *neural networks*, algoritma genetik, *machine learning*, robotika, dll

Keuntungan menggunakan *AForge.NET Framework* ini yaitu *class* yang disediakan oleh *AForge.NET Framework* dapat langsung digunakan sebagai API (*Application Programming Interface*) di *C#* program, sehingga membuat pengembangan aplikasi lebih cepat dan lebih mudah [11].

2.12.1 *Library AForge.NET*

Library pengolahan citra pada *AForge.NET Framework* memiliki satu set filter pengolohan citra dan dibangun untuk berbagai macam tugas yang ditujukan pada *computer vision* dan analisis/pengolahan citra. *Library* pada filter pengolahan citra sangat banyak dipakai maka perkembangannya pun semakin banyak dan baru. Contoh berbagai macam set filter pengolahan citra pada *library* pengolahan citra pada *AForge.NET Framework* diantaranya :

1. *Color filtering* (*grayscale*, sepia, *invert*, *rotate channels*, *channel extract*ion, *channel replacing*, *channel filtering*, *Color filtering*, *Euclidean Color filtering*, RGB *channels linear correction*).
2. HSL filter (*linear correction*, brightness, contrast, *Saturation*, *Hue* modifier, HSL *filtering*).
3. YCbCr *filter* (*linear correction*, YCbCr *filtering*, *channel extract*ion/replacement).
4. Deteksi tepi (*homogeneity*, *difference*, *sobel*, *canny*).
5. *Blob counter* dan *connected components labeling filter*.

Resize dan *rotation* (*nearest neighbor*, *bilinear*, *bicubic*

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan dari hasil penelitian, analisis sistem, perancangan sistem , dan pengujian sistem pada perangkat *virtual drawing* ini maka dapat diambil kesimpulan bahwa:

1. Penelitian ini telah berhasil mendeteksi dan *tracking* objek secara *realtime* dengan menggunakan metode *color filtering* yang di pakai pada penelitian sebelumnya.
2. Aplikasi *virtual drawing* dapat digunakan untuk menggambar secara virtual tanpa menggunakan mouse yang ditampilkan dalam bentuk *bitmap* yang dapat menghasilkan garis, bentuk bangun datar, dan bangun ruang.

5.2 Saran

Saran pembangunan aplikasi untuk pengembangan perangkat lunak ini adalah sebagai berikut:

1. Pendeteksian warna menggunakan *color filtering* yang diterapkan pada aplikasi ini masih bersifat statis hanya marker berwarna merah saja, diharapkan adanya fitur tambahan untuk mengambil warna secara dinamis.
2. Penelitian ini masih berupa simulasi program, maka diharapkan dalam penelitian selanjutnya dapat diterapkan secara nyata.

6. DAFTAR PUSTAKA

[1] U. Ahmad, Pengolahan Citra Digital dan Pemogramannya, Yogyakarta: Graha Ilmu,

2005.

[2] R. Munir, Pengolahan Citra Digital, Bandung: Informatika Bandung, 2004.

[3] Fadliansyah, Computer Vision dan Pengolahan Citra, Yogyakarta: ANDI, 2007.

[4] K. Nummiaro, E. Koller-Meier and L. V. Gool, "An Adaptive Color-Based Particle Filter," *Preprint submitted to Elsevier Science*, 2002.

[5] U. Umar, R. Soelistijorini and H. A. Darwito, "Tracking Arah Gerakan Telunjuk Jari Berbasis Webcam Menggunakan Metode Optical Flow," *Electronic Engineering Polytechnic Institute of Surabaya (EEPIS)*, 2011.

[6] A. Kirillov, "Lego Pan Tilt Camera and Objects Tracking," AForge.NET, 2008.

[7] S. J. Prince, Computer Vision: Models, Learning and Inference, Cambridge University Press, 2012.

[8] M. H. Purnomo and A. Muntasa, Konsep Pengolahan Citra Digital dan Ekstrasi Fitur, Yogyakarta: Graha Ilmu, 2010.

[9] A. Basuki, "Aplikasi Pengolahan Citra Deteksi Warna," PENS-ITS, Surabaya, 2009.

[10] G. S. Nahla, S. and D. Pramadihanto, "Tracking Bola Menggunakan Robotino".

[11] A. Kirillov, "AForge.NET Framework," AForge.NET, [Online]. Available: http://www.aforgenet.com/framework/. [Accessed October 2014].

[12] A. Nalwan, Web Kamera, Jakarta: Elex Media Komputindo, 2000.