

Real Time Tracking Object Moving With Webcam Based Color Using Background Subtraction Method

A.T. Jaka Harjanta¹, F.M. Dewanto²

¹ Pogram Studi Informatika Fakultas TEKNIK, Universitas PGRI Semarang
Gedung B Lantai 3, Kampus 1 Jl. Sidodadi Timur 24, Semarang, e-mail: aristrijaka@gmail.ac.id

² Pogram Studi Informatika Fakultas TEKNIK, Universitas PGRI Semarang
Gedung B Lantai 3, Kampus 1 Jl. Sidodadi Timur 24, Semarang, e-mail: aristrijaka@gmail.ac.id

ARTICLE INFO

ABSTRACT

Article history:

Received 02 Februari 2017
Received in revised form 29 June 2017
Accepted 24 July 2017
Available online 31 July 2017

The process of tracking objects in real time video is one of the important topics in the study of surveillance system (Dhananjaya, Rama, and Thimmaiah 2015). Detection and extraction of information and tracking of objects or moving objects is as one form of application from computer vision. Some applications that use the method of tracking objects or moving objects include UAV (Unmanned Aerial Vehicle) surveillance or better known as the engine / vehicle unmanned, Indoor Monitoring system is an indoor monitoring system, as well as monitoring traffic traffic that can observe the movement of all Objects in real time state. Tracking objects in real time state many things to note and need to be taken into account where all the parameters and noise or disturbance of the surrounding objects that we do not need to observe but still in one part with the object that we observe. In this research the method to be used is background subtraction For the detection of tracking object And moving objects in real time based on color By utilizing webcam cameras and using OpenCv opensource library.

Keywords: detection, object, color, video, opensource

1. Pendahuluan

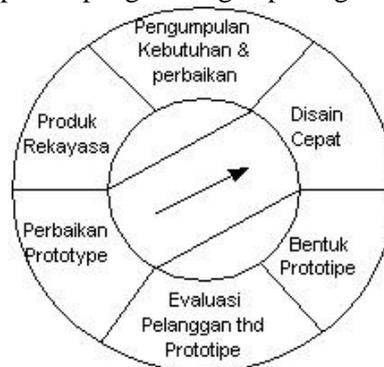
Dalam perkembangan teknologi informasi dan komunikasi modern muncul bidang yang mempelajari tentang *computer vision*. *Computer vision* merupakan salah satu bagian dari bidang teknologi informasi dan komunikasi yang merupakan perkembangan dari ilmu grafika serta pengolahan citra digital yang bergabung dengan berbagai ilmu bidang komputer yang di tujuan untuk meniru cara kerja dari pengelihat manusia yang dapat menangkap berbagai informasi diantaranya geometri, warna, ukuran, warna dan interpretasi dari suatu obyek.

Kajian di bidang *Computer vision* telah berkembang pesat di berbagai bidang diantaranya bagian militer, kesehatan, industri dan lain – lain dengan banyak ditemukannya peralatan yang canggih yang di hasilkan dari kajian *computer vision* antara lain yang menyangkut *biometric* seperti deteksi wajah orang, sidik jari, retina mata serta yang non biometric seperti pengenalan barang dengan sinar x, deteksi plat nomor kendaraan, jenis dan ukuran kendaraan serta banyak lagi hal lain yang ditemukan dari kajian mengenai *Computer Vision* (Szeliski 2010). Deteksi objek berdasarkan warna merupakan suatu kajian yang sangat menarik yang dapat di implementasikan kedalam berbagai kehidupan baik dunia industri misalnya dalam pendeteksian kematangan dan pensortiran buah-buahan berdasarkan warna, *quality control* secara otomatis pada industri kain, kertas dan sebagainya yang memerlukan pendeteksian berdasarkan warna adalah sangat banyak, meskipun tidak menutup kemungkinan dikembangkan pada industri sensorik

yang mengeluarkan instrumen – instrumen yang lebih spesifik dan *universal* untuk digunakan atau di integrasikan dengan peralatan peralatan lain.

2. Metode Penelitian

Metode penelitian yang digunakan adalah melakukan pengembangan perangkat lunak dengan menggunakan model prototyping. Model pengembangan perangkat lunak dengan model prototipe ini akan menghasilkan sebuah aplikasi dalam bentuk prototipe sebelum aplikasi tersebut memasuki tahap design. Dalam fase ini, prototype yang telah dirancangakan dievaluasi. Tahap ini akan terus menerus diulang sampai aplikasi benar benar sesuai dengan keinginan. Apabila prototipe telah selesai, maka tahapan aplikasi akan kembali berlanjut ketahap design. Gambar 2.1 menjelaskan bagaimana urutan proses pengembangan perangkat lunak dengan model prototyping.



Gambar 1. metode prototipe

Secara lebih lengkap gambar 2.1 dapat dijelaskan sebagai berikut :

a. Pengumpulan Kebutuhan

Pengumpulan kebutuhan adalah proses pendefinisian secara objektif mengenai Sistem Informasi yang akan dibangun, mengidentifikasi kebutuhan input dan output. Langkah yang dilakukan dalam mengidentifikasi input adalah dengan melakukan studi awal mengenai video dan bahasa pemrograman yang memungkinkan untuk digunakan untuk menjalankan sistem pendeteksi. Sedangkan output yang didapatkan adalah adanya sebuah sistem untuk yang dapat digunakan untuk mendeteksi sebuah object yang telah di tentukan dengan berdsarkan warna.

b. Perancangan Kilat

Perancangan kilat yang dilakukan adalah melakukan desain rancangan alur sistem yang ditawarkan, rancangan basis data dan rancangan antar muka serta rancangan alur program yang diakan digunakan.

c. Membangun Prototipe

Proses membangun prototipe adalah bentuk implementasi kedalam bahasa pemrograman dari hasil perancangan kilat.

d. Evaluasi Prototipe

Tahap evaluasi prototipe adalah proses dimana hasil pembuatan prototipe disesuaikan dengan kebutuhan, pada tahap ini dimungkinkan untuk melakukan prancangan ulang sehingga dihasilkan bentuk prototipe yang menghasilkan output sesuai dengan kebutuhan.

e. *Engineer Product*

Tahap engineer product adalah bentuk akhir dari prototipe yang dibangun yang telah sesuai dengan kecukupan kebutuhan.

3. Hasil dan Analisis

3.1 Bahan Penelitian

Bahan penelitian yang digunakan untuk membangun sebuah sistem *tracking* obyek secara *real time* yang dapat memisahkan object tertentu yang di tentukan berdasarkan warna dengan menggunakan metode *background subtraction* adalah sebagai berikut.

1) Input

Bahan inputan yang dimaksudkan didalam penelitian ini adalah video secara *real time* yang menampilkan adanya boyek tertentu yang dapat digunakan sebagai inputan sistem yang daam penelitian ini menggunakan inputan *webcam* dengan 720p FaceTime HD camera dari apple.inc.

2) Proses

Yang dimaksud proses disini adalah penggunaan bahasa pemrograman dan *libraries* program untuk dapat di gabungkan dengan menjadi sebuah sistem yang mampu memproses *input* untuk dijadikan sebuah *output* sesuai yang diharapkan, yaitu sebuah object yang sudah tersegmentasi dengan metode *background subtraction*. *Background Subtraction*, yang juga dikenal sebagai *Foreground Detection*, adalah salah satu teknik pada bidang pengolahan citra dan computer vision yang bertujuan untuk mendeteksi/mengambil foreground dari background untuk diproses lebih lanjut (seperti pada proses object recognition dll). *Background subtraction* merupakan metode yang umumnya digunakan untuk mendeteksi objek bergerak pada video dari kamera statis (*stationary camera*). Proses deteksi objek bergerak dengan metode *background subtraction* didasarkan pada perbedaan antara background referensi dengan frame. Dalam penelitian ini menggunakan Bahasa pemrograman python dan *libraries OpenCv*.



Gambar 2. Logo Bahasa Pemrograman Python dan Libraries Opencv

Untuk dapat menggabungkan instrumen bahasa pemrograman dan *libraries OpenCv* pada penelitian ini menggunakan komputer dengan sistem operasi Mac OS X El CAPITAN version 10.11.5 yang notabene berisi sistem operasi berbasis *linux (open source)* dengan processor 2,7 Ghz Intel Core i5 dengan meory 8 GB 1867 MHz DDR3 dan kartu grafis *Intel Iris Graphics* 6100 1536 MB.

3) Output

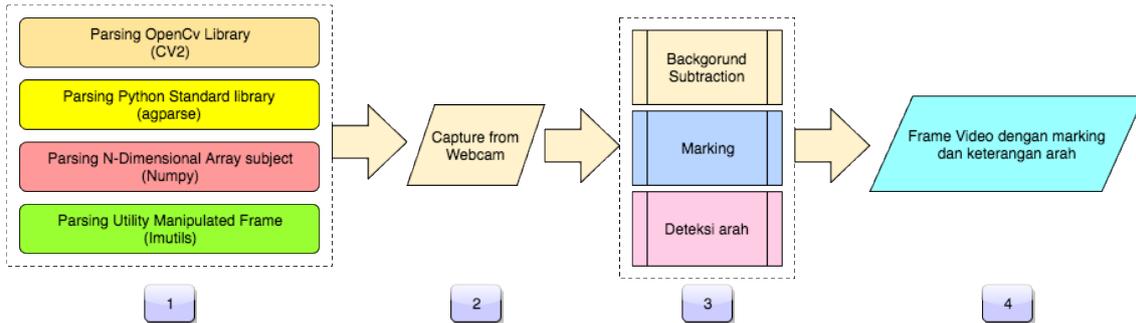
Output atau keluaran sistem disini yang dimaksud adalah keluaran sistem pendeteksi object yang telah tersegmentasi berdasarkan metode *background subtraction* dalam hal ini *output* di desain berupa tampilan frame yang berisi tapilan secara realtime dari obyek yang di *capture* oleh sebuah webcam dan ditampilkan dalam bentuk langsung dengan *marking* dengan pewarnaan pada obyek yang di *tracking* tersebut dan di berikan tanda yang menunjukka arah pergerakan obyek tersebut, serta di berikan keterangan arah pergerakanya

3.2 Analisa Sistem Berjalan

Dalam proses yang sedang berjalan, proses dimana video masukan/*input* dapat di olah oleh sistem dan diproses untuk dapat di tentukan bagian tertentu yang sesuai dengan setting warna dan di pisahkan dengan bagian latar belakang nya. Dan kemudian mensegmentasi bagian tersebut. Setelah itu sistem menggabungkan bagian dari video asli dengan bagian *threshold* dan disertai arah pergerakan obyek.

3.3 Rancangan sistem

Rancangan sistem pendeteksi obyek ini dapat digambarkan dengan bagan sebagai berikut:



Gambar 3. Rancangan Sistem Deteksi Obyek

keterangan proses

1. Proses dimana sistem memuat dan menginisialisasi sistem *librariesOpenCV*, *libraries python*, *N-dimensional array*, dan *utility manipulated frame*
2. Proses memasukkan inputan berupa *capture* dari webcam
3. Konversi tiap tiap *frame* dari video lalu diproses dengan *background subtraction* untuk memisahkan obyek yang di deteksi dan diberikan *marking* dan dideteksi arah pergerakan dari *marking* tersebut.
4. Mengeluarkan / *view* berupa fram video dengan disertai *marking* dan keterangan arah pergerakan obyek.

3.4 Hasil Implementasi

Pengujian merupakan bagian yang penting dalam pembangunan sebuah perangkat lunak, pengujian ditujukan untuk menemukan kesalahan-kesalahan pada sistem dan memastikan sistem yang dibangun telah sesuai dengan apa yang direncanakan sebelumnya. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal, yaitu mampu mempresentasikan kajian pokok dari spesifikasi analisis, perancangan dan pengkodean dari perangkat lunak itu sendiri.

Pengujian Rancangan pengujian yang akan dilakukan dalam pembangunan sistem ini adalah dengan menggunakan metode pengujian black box. Pengujian black box ini menitikberatkan pada fungsi sistem. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar.

a. Pengujian Deteksi Object

Tahapan ini adalah menguji sistem dalam mendeteksi object sesuai yang di sesuaikan dengan konfigurasi pada sistem

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Benda berwarna sesuai dengan konfigurasi	Sistem Menampilkan	tampil gambar <i>object</i> dengan	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai

	Gambar <i>Object</i> dengan disertai <i>Marking</i> berwarna merah muda	disertai <i>Marking</i>	
Kasus dan Hasil Uji (Data Tidak Normal)			
Benda berwarna tidak sesuai dengan konfigurasi	Sistem Menampilkan Gambar <i>Object</i> dengan tidak disertai <i>Marking</i> berwarna merah muda	tampil gambar <i>object</i> dengan tidak disertai <i>Marking</i>	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai

b. Pengujian *Tracking Object*

Tahapan ini adalah menguji sistem dalam mengikuti/ *tracking object* sesuai yang di sesuaikan dengan konfigurasi pada sistem

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pergerakan benda / object berwarna sesuai dengan konfigurasi	Sistem Menampilkan pergerakan <i>Object</i> dengan disertai <i>Marking</i>	tampil pergerakan gambar <i>object</i> dengan disertai <i>Marking</i>	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai
Kasus dan Hasil Uji (Data Tidak Normal)			
Pergerakan benda / object berwarna tidak sesuai dengan konfigurasi	Sistem Menampilkan pergerakan <i>Object</i> dengan tidak disertai <i>Marking</i>	tampil pergerakan gambar <i>object</i> dengan tidak disertai <i>Marking</i>	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai

c. Pengujian Arah *Tracking Object*

Tahapan ini adalah menguji sistem dalam mengikuti / *tracking object* dengan deteksi arah sesuai yang di sesuaikan dengan konfigurasi pada sistem

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pergerakan ke arah Utara pada benda / object berwarna sesuai dengan konfigurasi	Sistem Menampilkan pergerakan <i>Object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah utara	tampil pergerakan gambar <i>object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah utara	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai
Pergerakan ke arah Selatan pada benda / object berwarna sesuai dengan konfigurasi	Sistem Menampilkan pergerakan <i>Object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Selatan	tampil pergerakan gambar <i>object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Selatan	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai
Pergerakan ke arah Timur pada benda / object berwarna sesuai dengan konfigurasi	Sistem Menampilkan pergerakan <i>Object</i> dengan disertai <i>Marking</i> dan keterangan menuju	tampil pergerakan gambar <i>object</i> dengan disertai <i>Marking</i> dan keterangan menuju	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai

	arah Timur	arah Timur	
Pergerakan ke arah Barat pada benda / object berwarna sesuai dengan konfigurasi	Sistem Menampilkan pergerakan <i>Object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Barat	tampil pergerakan gambar <i>object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Barat	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai
Pergerakan ke arah Serong ke Tenggara (Selatan - Timur) pada benda / object berwarna sesuai dengan konfigurasi	Sistem Menampilkan pergerakan <i>Object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Selatan - Timur	tampil pergerakan gambar <i>object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Selatan - Timur	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai
Pergerakan ke arah serong ke arah Barat daya (Selatan – Barat) pada benda / object berwarna sesuai dengan konfigurasi	Sistem Menampilkan pergerakan <i>Object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Selatan – Barat	tampil pergerakan gambar <i>object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Selatan – Barat	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai
Pergerakan ke arah serong ke arah Barat Laut (Utara – Barat) pada benda / object berwarna sesuai dengan konfigurasi	Sistem Menampilkan pergerakan <i>Object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Utara – Barat	tampil pergerakan gambar <i>object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Utara – Barat	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai
Pergerakan ke arah serong ke arah Timur Laut (Utara – Timur) pada benda / object berwarna sesuai dengan konfigurasi	Sistem Menampilkan pergerakan <i>Object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Utara – Timur	tampil pergerakan gambar <i>object</i> dengan disertai <i>Marking</i> dan keterangan menuju arah Utara – Timur	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai
Kasus dan Hasil Uji (Data Tidak Normal)			
Benda / object yang tidak bergerak	Sistem Menampilkan <i>Object</i> dengan tidak disertai keterangan arah	tampil gambar <i>object</i> dengan tidak disertai keterangan pergerakan	[<input checked="" type="checkbox"/>] Sesuai [<input type="checkbox"/>] Tidak Sesuai

4. Kesimpulan dan Saran

4.1 Kesimpulan

Berdasarkan hasil penelitian, dengan menggunakan *webcam* untuk *tracking* obyek warna yang bergerak secara *real time* dengan menggunakan metode *Background Subtraction* maka didapatkan kesimpulan sebagai berikut :

- Webcam* mampu menangkap *frame* gambar atau video secara *realtime* dengan baik
- Dengan metode *background substarction* dapat dipisahkan warna yang telah di seleksi dengan latar belakang nya.
- Dapat ditambahkan penanda (*marker*) dengan bentuk bulatan warna pada object dengan warna yang telah di tentukan.
- Dengan marker tersebut dapat ditentukan kemanapun arah object tersebut bergerak secara *real time*.

4.2 Saran

Sehubungan dengan hasil penelitian ini, peneliti ingin menyampaikan beberapa saran yang mungkin berguna bagi penelitian berikutnya sebagai berikut:

- a. Penambahan resolusi gambar atau video.
- b. Penambahan variasi obyek dan warna secara simultan.
- c. Pengukuran akurasi *tracking* dengan kecepatan pergerakan obyek

Referensi

- [1] Bedford, Virginia. "Use of Publicly Available Webcams in Naturalistic Observation Studies." *dataprivacylab.org*: 1–3.
- [2] Cheung, Sen-ching S., and Chandrika Kamath. "Robust Techniques for Background Subtraction in Urban Traffic Video". Sethuraman Panchanathan and Bhaskaran Vasudev. 2004 : 881–92.
- [3] Dhananjaya, B. "Detection of Objects in Aerial Videos for Object Extraction and Tracking for UAV Applications." . 2015: 112(12): 37–42.
- [4] Dhananjaya, B, B Rama, and P Thimmaiah. "Moving Object Tracking with OpenCV on ARM Cortex-A8 in Surveillance Applications" . 2015: 5(2). 843–48.
- [5] Gonzalez, Rafael C, Richard E Woods, Art Heather Scott, and Pearson Prentice Hall. *Digital Image Processing*. 2008.
- [6] Jacobs, Nathan, Kyla Miskell, and Robert Pless. "Webcam Geo-Localization Using Aggregate Light Levels".2010: 132–38.
- [7] Kluge, B., C. Kohler, and E. Prassler. "Fast and Robust Tracking of Multiple Moving Objects with a Laser Range Finder". *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*. 2001: 2. 1683–88.
- [8] Piccardi, M. "Background Subtraction Techniques: A Review". *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*. 2004: 3099–3104.
- [9] Soeleman, Moch Arief, Ricardus Anggi P, and Pulung Nurtantio Andono. "Background Subtraction Berbasis Algoritma K-Means Klustering Untuk Deteksi Objek Bergerak". 2014 : 246–49.
- [10] Szeliski, Richard. "Computer Vision : Algorithms and Applications". 2010.
- [11] Yin, Fei, Dimitrios Makris, and Sergio Velastin. "Performance Evaluation of Object Tracking Algorithms."