**BAB II**

**TINJAUAN PUSTAKA**

* 1. **DASAR PERMASALAHAN**

Selama ini sudah banyak alat atau sensor yang bisa digunakan untuk memperkirakan kecepatan sebuah kendaraan, contohnya adalah radar/*speed detector*, *detector microwave*, *sensor tube* dan *detector loop.* Secara umum, kekurangan dari penggunaan sensor-sensor tersebut adalah pemasangan yang lebih rumit dan juga biaya yang dibutuhkan cenderung tinggi.

Berbeda jika menggunakan kamera sebagai pengganti sensor-sensor tersebut. Sensor berbasis kamera lebih baik dibandingkan sensor-sensor lain yang digunakan selama ini. Dari segi instalasi, biaya yang diperlukan juga lebih sedikit dan jauh lebih mudah pemasangannya. Sebuah kamera bisa memonitor lebih dari satu jalur jalan. Jangkauannya bisa lebih jauh hingga ratusan meter tergantung letak dan jenis kamera yang digunakan. Informasi yang didapat dari sensor berbasis kamera ini juga lebih banyak, yaitu gambar dan kecepatan kendaraan. Oleh karena itu, untuk sekarang ini penggunaan kamera sebagai sensor kecepatan kendaraan lebih disarankan daripada penggunaan sensor-sensor lainnya.

* 1. **PENELITIAN TERKAIT**

Penelitian yang berkaitan dengan proyek akhir ini adalah sebagai berikut :

Ubaidillah Umar, Reni Soelistijorini, Haryadi Amran Darwito, dalam proyek akhir *Tracking Arah Gerakan Telunjuk Jari Berbasis Webcam Menggunakan Metode Optical Flow*, Teknik Telekomunikasi – Politeknik Elektronika Negeri Surabaya, 2011[1] menyatakan,

“Deteksi gerakan dengan melakukan tracking arah gerakan telunjuk jari berbasis webcam dengan menggunakan Metode *Optical Flow*. Webcam terhubung dengan komputer server untuk melakukan proses capture kondisi ruangan atau pengambilan gambar dan disimpan dalam file .jpg pada periode waktu tertentu. Dengan menggunakan metode *Optical Flow*, webcam akan mendeteksi dan mentracking jika ada arah gerakan telunjuk jari. Berdasarkan hasil penelitian, dari 120 percobaan yang telah dilakukan 117 diantaranya dinyatakan baik sehingga dapat diambil nilai keberhasilan penelitian sebesar 97,5%.”[1]

Dimas Arioputra, dalam skripsi *Analisa Perbandingan Akurasi Metode Optical Flow Dan Gaussian Mixture Model untuk Sistem Pemantau Lalu Lintas Berbasis Computer Vision*, Teknik Komputer – Universitas Indonesia, 2012[2] menyatakan,

“Teknik yang digunakan untuk sistem pemantau lalu lintas pada masa sekarang ini banyak tergantung pada sensor-sensor yang mempunyai kemampuan yang terbatas, kurang fleksibel, dan seringkali mahal dan sulit untuk dipasang. Penggunaan kamera digabungkan dengan teknologi *Computer Vision* menjadi alternatif yang menarik dari sensor yang ada saat ini. Sensor berbasis kamera ini mempunyai potensi yang lebih besar untuk mengamati kondisi lalu lintas yang ada dibanding sensor konvensional saat ini yaitu sensor ini lebih murah dan mudah untuk dipasang.Di dalam penelitian ini dikembangkan sistem pemantau lalu lintas menggunakan metode *Optical Flow* dan *Gaussian Mixture Model*. Eksperimen dilakukan menggunakan *handycam*, berlokasi di salah satu tol dalam kota Jakarta. Kondisi pengambilan gambar adalah pada kondisi yang berbeda beda yaitu pada saat pagi, siang, dan sore, cuaca cerah, dan mendung, serta kondisi arus lalu lintas padat dan lancar. Setelah pengujian dilakukan, algoritma *Optical Flow* memberikan hasil yang lebih baik dibandingkan algoritma *Gaussian Mixture Model* yaitu dengan akurasi mencapai 92% dibanding *Gaussian* yang hanya mencapai 72%. Faktor-faktor yang mempengaruhi tingkat akurasi adalah kondisi waktu, cuaca, dan arus kendaraan serta lokasi pengambilan gambar.”[2]

* 1. **DASAR TEORI**

Pembatasan kecepatan adalah suatu ketentuan untuk membatasi kecepatan lalu lintas kendaraan dalam rangka menurunkan angka kecelakaan lalu-lintas. Untuk membatasi kecepatan ini digunakan aturan yang sifatnya umum ataupun aturan yang sifatnya khusus untuk membatasi kecepatan yang lebih rendah karena alasan keramaian, di sekitar sekolah, banyaknya kegiatan di sekitar jalan, penghematan energi ataupun karena alasan geometrik jalan.

Kurang lebih sepertiga korban kecelakaan yang meninggal karena pelanggaran kecepatan, sehingga pembatasan kecepatan merupakan alat yang ampuh untuk mengendalikan jumlah korban yang meninggal akibat kecelakaan lalu-lintas.

Untuk itu sebaiknya kita mematuhi batas kecepatan kendaran anda apabila memasuki beberapa kawasan. Batas laju kendaraan akan memperkecil resiko kecelakaan yang dapat menimbulkan korban jiwa maupun materil. Secara umum, pembatasan kecepatan kendaraan diatur dalam pasal 80 Peraturan Pemerintah no. 43 tahun 1993 tentang Prasarana dan Lalu Lintas Jalan. Dilihat dari kelas jalannya, batas kecepatan bisa dilihat pada tabel dibawah ini [3]:

|  |  |  |  |
| --- | --- | --- | --- |
| **Kelas**  **Jalan** | **Fungsi** | **Jenis**  **Kendaraan** | **Batas Kecepatan (Km/Jam)** |
| Kelas I, II dan IIIA | Primer | Mobil Pribadi, Bus, Truk | 100 |
| Kelas I, II dan IIIA | Primer | Gandengan Dan Tempelan | 80 |
| Kelas IIIB | Primer | Mobil Pribadi, Bus, Truk | 80 |
| Kelas IIIC | Primer | Mobil Pribadi, Bus, Truk | 60 |
| Kelas II, IIIA | Sekunder | Mobil Pribadi, Bus, Truk | 70 |
| Kelas II, IIIA | Sekunder | Gandengan, Tempelan | 60 |
| Kelas IIIB | Sekunder | Mobil Pribadi, Bus, Truk | 50 |
| Kelas IIIC | Sekunder | Mobil Pribadi, Bus, Truk | 40 |

**Tabel 2.1.** Batas Kecepatan Kendaraan

Selain berdasarkan kelas, pembatasan kecepatan juga dilakukan berdasarkan desain jalan tersebut atau bisa disebut dengan kecepatan rencana. Yang dimaksud dari kecepatan rencana atau *design speed* adalah kecepatan kendaraan yang dapat dicapai bila berjalan tanpa gangguan dan aman. Jalan dengan kecepatan rencana paling rendah 60 km/jam adalah jalan yang didesain dengan persyaratan- persyaratan *geometric* yang diperhitungkan terhadap kecepatan minimum 60 km/jam, sehingga pada volume jam perencanaan (*design hourly volume*) kendaraan bermotor dapat menggunakan kecepatan 60 km/jam dengan aman.

Persyaratan kecepatan rencana diambil angka paling rendah dengan maksud untuk memberikan kebebasan bagi perencana jalan dalam menetapkan kecepatan rencana yang paling tepat, disesuaikan dengan kondisi lingkungannya, sama atau lebih besar dari persyaratan tersebut, agar dicapai kapasitas jalan yang paling tinggi.

Semakin tinggi kecepatan rencana ditetapkan semakin mahal biaya untuk pembangunan jalannya karena dibutuhkan radius tikung yang semakin besar dan tanjakan/turunan yang semakin kecil, jalan tol biasanya direncanakan pada kecepatan rencana yang tinggi yaitu 100 km/jam. Untuk menyampaikan informasi kecepatan rencana kepada pengguna jalan digunakan rambu lalu lintas batas kecepatan.

Pertimbangan yang digunakan dalam penetapan kecepatan rencana antara lain:

* Biaya pembangunan jalan
* Medan yang dilalui
* Klasifikasi jalan
* Besarnya perkiraan arus lalu lintas
* Keselamatan, semakin rendah batas kecepatan semakin tinggi keamanan pengguna jalan
* Penggunaan energy

Kecepatan rencana ditetapkan berdasarkan klasifikasi jalan dan medan yang dilalui, di medan jalan yang datar akan lebih tinggi dari perbukitan ataupun pegunungan. Besaran kecepatan rencana di Indonesia bisa dilihat pada tabel dibawah ini [4] :

|  |  |  |  |
| --- | --- | --- | --- |
| **Kelas** | **Datar**  **(Km/Jam)** | **Bukit**  **(Km/Jam)** | **Gunung**  **(Km/Jam)** |
| Jalan Tol  (Arteri & kolektor) | 100 – 120 | 80 – 100 | 80 |
| Jalan Raya  (Arteri & Kolektor) | 80 – 100 | 60 – 80 | 40 – 60 |
| Jalan Sedang  (Kolektor) | 60 – 80 | 40 – 60 | 40 |
| Jalan Kecil  (Lokal & Lingkungan) | 20 – 40 | 20 – 40 | 20 – 40 |

**Tabel 2.2.** Batas Kecepatan Rencana

* 1. **TEORI PENUNJANG**
     1. **Metode Optical Flow**

*Optical Flow* adalah perkiraan gerakan suatu bagian dari sebuah citra berdasarkan turunan intensitas cahayanya pada sebuah sekuen citra. Pada ruang 2D hal ini berarti seberapa jauh suatu piksel citra berpindah diantra dua frame citra yang berurutan. Sedangkan pada ruang 3D hal ini berarti seberapa jauh suatu volume piksel (voxel) berpindah pada dua volume yang berurutan. Perhitungan turunan dilakukan berdasarkan perubahan intensitas cahaya pada kedua frame citra maupun volume. Perubahan intensitas cahaya pada suatu bagian citra dapat disebabkan oleh gerakan yang dilakukan oleh obyek, gerakan sumber cahaya, ataupun perubahan sudut pandang.[1]

Metode optical flow bekerja dengan cara meghitung gerak antara dua frame gambar yang diambil pada waktu **t** dan **t+∆t** pada setiap posisi volume piksel (voxel). Di asumsikan **I(x,y,t)** adalah titik tengah **n x n** dari lokasi dimana ada objek bergerak dengan **∆x**, **∆y** dalam waktu **∆t** ke **I(x+∆x,y+∆y,t+∆t)**. Karena **I(x,y,t)** dan **I(x+∆x,y+∆y,t+∆t)** adalah gambar dari titik yang sama, maka bisa dirumuskan :

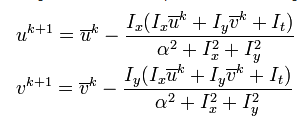
**I(x,y,t)** = **I(x+∆x,y+∆y,t+∆t)** (1)

Dari persamaan (1) bisa dipecahkan menjadi :

**IxVx + IyVy = -It** (2)

Dimana **Ix, Iy, It** adalah intensitas derivatif pada masing-masing **x,y,t** dan **Vx, Vy** adalah komponen **x** dan **y** dari kecepatan atau *Optical Flow* **I(x,y,t)**.

Persamaan (2) kemudian bisa diselesaikan dengan menggunakan metode Horn-Schunck :

 (3)

Dimana **Ix, Iy** dan **It** merupakan turunan dari intensitas citra ,  merupakan *Optical Flow* horizontal,  merupakan *Optical Flow* vertical α merupakan konstanta *smoothness*,  merupakan perkiraan kecepatan sebuah piksel pada (x,y) dan  merupakan rata-rata kecepatan pada titik-titik sebelumnya. Dengan menggunakan algoritma tersebut, maka dapat dicari vector pergerakan dari benda yang ingin dilacak. Dalam tugas akhir ini objeknya adalah kendaraan yang dianalisa vector pergerakannya.

* + 1. **Threshold**

*Thresholding* adalah proses mengubah citra berderajat keabuan menjadi citra biner atau hitam putih sehingga dapat diketahui daerah mana yang termasuk obyek dan *background* dari citra secara jelas. Citra hasil *thresholding* biasanya digunakan lebih lanjut untuk proses pengenalan obyek serta ekstraksi fitur. Metode *thresholding* secara umum dibagi menjadi dua, yaitu :

1. *Thresholding* global

*Thresholding* dilakukan dengan mempartisi histogram dengan menggunakan sebuah threshold (batas ambang) global T, yang berlaku untuk seluruh bagian pada citra.

1. *Thresholding* adaptif

*Thesholding* dilakukan dengan membagi citra menggunakan beberapa sub citra. Lalu pada setiap sub citra, segmentasi dilakukan dengan menggunakan *threshold* yang berbeda.[5]

Secara matematis ditulis seperti berikut :

**** (4)

Dengan g(x,y) adalah citra asli (input), f(x,y) adalah piksel citra baru (hasil/output), T adalah nilai ambang yang ditentukan. Nilai piksel pada (x,y) citra output akan sama dengan 1 jika nilai piksel (x,y) citra input tersebut > T. Nilai piksel (x,y) citra output akan sama dengan 0 jika nilai piksel (x,y) < T.

* + 1. **BLOB *Analysis***

Algoritma ini digunakan untuk mendeteksi titik atau area pada sebuah citra yang berbeda baik warna ataupun kecerahannya jika dibandingkan dengan sekitarnya. Algoritma ini banyak digunakan untuk mendapatkan informasi yang tidak dapat diperoleh jika menggunakan algortima *edge detection* atau *corner detection* . Dengan demikian informasi yang diperoleh menjadi sangat berguna untuk pendeteksian objek. Berikut adalah metode untuk mencari BLOB menggunakan *Laplacian of Gaussian*. Persamaannya yang menyatakan hubungan sederhana dari *Laplacian of Gaussian* seperti dibawah ini :

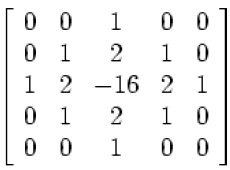
**LoG \* f (x,y)** (5)

Dimana f(x,y) merupakan bagian piksel yang diberi fungsi *Laplacian of Gaussian*.



**Gambar 2.1.** BLOB Analysis

Sebuah BLOB dapat dideteksi dengan melihat maksima dari konvolusi *Laplacian of Gaussian.* Maksima dapat dilihat dari distribusi *Laplacian of Gaussian* seperti yang ditunjukkan pada gambar 2.1 dibawah ini :

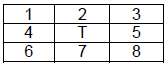


**Gambar 2.2.** Matriks Hasil *Laplacian of Gaussian*

Matriks *Laplacian of Gaussian* diberi proses *threshold* untuk menguatkan bagian mana yang positif dan bagian mana yang negatif sehingga BLOB citra dapat diketahui.[2]

* + 1. **Mean Fitering**

Mean filtering yang digunakan untuk efek smoothing ini merupakan jenis *spatial filtering*, yang dalam prosesnya mengikutsertakan piksel-piksel disekitarnya. Piksel yang akan diproses dimasukkan dalam sebuah matrik yang berdimensi N x N. Ukuran N ini tergantung pada kebutuhan, tetapi nilai N haruslah ganjil sehingga piksel yang diproses dapat diletakkan tepat ditengah matrik. Sebagai contoh matrik berdimensi 3 x 3 seperti gambar 2.3 di bawah ini:



**Gambar 2.3.** Matriks *Mean Filtering*

Nilai 1, 2, 3, 4, 5, 6, 7, dan 8 pada gambar 2.1 adalah piksel-piksel disekitar piksel T yang akan diproses. Nilai 4 didapat dari piksel sebelah kiri dari piksel T, nilai 5 didapat dari piksel di sebelah kanan dari piksel T, proses pengambilan piksel dimulai dengan mengambil piksel yang akan diproses, disimpan dalam nilai T. Kemudian diambil piksel-piksel sekitarnya sehingga matrik terisi penuh. Proses selanjutnya dijumlahkan semua nilai yang terdapat pada matrik tersebut. Hasil penjumlahan tersebut dibagi dengan jumlah titik yang terdapat pada matrik tersebut. Bilangan pembagi ini dapat diperoleh dari perkalian antara N x N. Pada gambar 2.3, maka hasil pembaginya adalah 9. Sembilan diperoleh dari hasil kali matrik 3 x 3. Hasil pembagian tersebut akan menggantikan nilai T. Nilai T yang baru akan ditampilkan pada hasil *threshold* untuk menggantikan nilai T yang lama. [7]

* + 1. **Adaptive Thresholding**

*Adaptive thresholding* atau disebut juga *thresholding* local berfungi untuk menentukan nilai *threshold* setiap piksel berdasarkan nilai grayscale sendiri dan nilai grayscale tetangga. Oleh karena itu, pendekatan ini disebut algoritma *thresholding* adaptif. Beberapa algoritma *thresholding* adaptif telah dikembangkan, diantaranya adalah metode Niblack pada tahun 1986. Namun, metode Niblack tidak efektif ketika background berisi tekstur berupa cahaya. Pada tahun 2000, Sauvola menyusun metode yang memodifikasi metode Niblack dengan melakukan hipotesis pada nilai keabuan piksel teks dan piksel background. Pada tahun 2004, Sezgin dan Sankur melakukan perbandingan terhadap 40 metode. Hal itu menunjukkan bahwa metode thresholding lokal Sauvola dan metode dari Trier melakukan yang terbaik untuk proses thresholding dokumen.[8]

Oleh karena itu, adalam proyek akhir ini jenis *adaptive thresholding* yang digunakan adalah sauvola. Pada metode sauvola, nilai *threshold T*(x,y) dihitung menggunakan rata-rata m(x,y) dan standart deviation d(x,y) . Dari sebuah matriks berukuran N x N cara menghitungnya adalah :

 (6)

Dimana R adalah nilai maksimum dari *standart deviation* (R = 128 untuk gambar *grayscale*) dan k adalah bias dimana memiliki nilai positif dalam range 0.2 – 0.5.[9]

* + 1. **Estimasi Kecepatan**

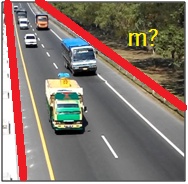
Dalam melakukan estimasi terhadap kecepatan kendaraan dalam video, proses yang harus dilakukan yaitu :

1. **Penghitungan Jarak**

Jarak merupakan salah satu komponen penting untuk bisa menentukan berapa kecepatan kendaraan dalam video atau sekuen gambar. Tetapi masalahnya, sekuen gambar merupakan sebuah gambar 2D sehingga tidak bisa diketahui secara pasti jarak sebenarnya dalam sekuen gambar tersebut. Ada beberapa cara untuk bisa mengetahui jarak sebenarnya dari sekuen gambar, diantaranya :

1. **Mengukur Panjang Lintasan Yang Terekam Dalam Video**

Hal ini susah dilakukan, salah satu alasannya karena berbahaya. Untuk mengukur panjang lintasan kita harus turun langsung ke jalan. Selain itu, jika video diambil dari tempat berbeda maka harus mengukur lagi panjang lintasannya. Sebenarnya bisa diatasi dengan mengatur tinggi kamera saat merekam dan arah perekamannya, tapi sulit untuk di implementasikan karena tinggi jembatan penyebrangan tidak selalu sama.



**Gambar 2.3.** Simulasi Hitung Panjang Lintasan

1. **Menggunakan Panjang Kendaraan Sebagai Acuan**

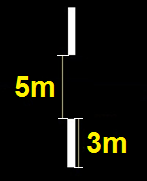
Hal ini lebih mudah daripada cara pertama (a). Tetapi kadang beberapa kendaraan mempunyai panjang yang berbeda, bisa dikarenakan adanya modifikasi dari pemilik kendaraan. Selain itu, kita juga harus mempunyai data mengenai panjang sebenarnya dari kendaraan yang kita jadikan acuan.



**Gambar 2.4.** Simulasi Hitung Panjang Kendaraan

1. **Menggunakan Marka Jalan Sebagai Acuan**

Hal ini lebih mudah dilakukan daripada dua cara sebelumnya. Sesuai dengan keputusan menteri perhubungan nomor KM 60 Tahun 1993, setiap jalan tol memiliki panjang marka jalan putus-pustus yang sama, yaitu panjang marka 3m dan jarak antar marka 5m.[10]

****

**Gambar 14**. Panjang Marka Jalan Tol

Dengan menggunakan acuan marka jalan (c), kita bisa mengetahui berapa panjang sebenarnya sebuah jalan dari video yang akan diolah. Sehingga bisa mengetahui panjang sebenarnya dari setiap pixel dalam sekuen gambar.

1. **Penghitungan Kecepatan**

Setelah mengetahui jarak sebenarnya yang ditempuh oleh kendaraan, selanjutnya adalah menghitung berapa kecepatan dari kendaraan tersebut. Rumus untuk menghitung kecepatan adalah :

v = (6)

Dimana d (dalam m) adalah jarak yang ditempuh oleh kendaraan. Sedangkan t adalah waktu yang diperlukan kendaraan untuk menempuh jarak (d).

* 1. **ORIGINALITAS PENELITIAN**

Proyek akhir ini menggunakan sebuah pendekatan baru untuk membuat sebuah sistem yang dapat memperkirakan kecepatan sebuah kendaraan menggunakan sensor kamera. Hasil dari sistem ini bukan hanya kecepatannya kendaraan saja, melainkan juga berupa gambar kendaraan apabila kendaraan tersebut melewati batas kecepatan yang ditentukan. Sistem ini juga dilengkapi *report* untuk mengetahui statistik jumlah pelanggar dan rata-rata kecepatan pelanggar. Proyek ini dibuat menggunakan Delphi dan tanpa menggunakan bantuan library *image processing* seperti OpenCV, Matlab dan library lainnya.