

## Pemrosesan Video Pendeteksi Kecepatan dan Ketinggian Aliran Lahar Dingin Pendukung Sistem Peringatan Dini

Lukman Awaludin<sup>\*1</sup>, Agus Harjoko<sup>2</sup>, Raden Sumiharto<sup>3</sup>

<sup>1</sup>Prodi Elektronika dan Instrumentasi FMIPA, UGM Yogyakarta

<sup>2,3</sup>Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: \*[awal.elins@gmail.com](mailto:awal.elins@gmail.com), [aharjoko@ugm.ac.id](mailto:aharjoko@ugm.ac.id), [r\\_sumiharto@ugm.ac.id](mailto:r_sumiharto@ugm.ac.id)

### Abstrak

*Pemrosesan video pendeteksi kecepatan dan ketinggian aliran lahar dingin pendukung sistem peringatan dini merupakan salah satu sistem yang memanfaatkan computer vision system untuk deteksi kecepatan dan ketinggian aliran lahar dingin, yang selama ini belum menggunakan sensor deteksi kecepatan dan ketinggian lainnya karena aliran lahar dingin bersifat merusak. Sistem ini merupakan tahap awal dalam perkembangan kearah sistem peringatan dini.*

*Metode deteksi ini dirancang untuk dapat mendeteksi kecepatan dan ketinggian aliran lahar dingin dengan menggunakan metode optical flow pyramidal Lucas Kanade, edge detection Sobel, thresholding, frame adder, hough transform, dilation dan Region of Interest. Pemrograman menggunakan OpenCV 2.3.1 dan menggunakan Visual Studio 2010. Bahasa pemrograman yang digunakan adalah bahasa C++.*

*Pengujian dengan beberapa sampel keadaan menunjukkan bahwa metode ini memiliki nilai standar deviasi sebesar 0,033 untuk deteksi kecepatan, dan nilai standar deviasi 0,035 untuk deteksi ketinggian aliran. Kecepatan minimal dan maksimal yang dapat terdeteksi bergantung pada jenis kamera yang digunakan dalam pengambilan data video, dalam hal ini sistem dengan kamera yang digunakan dapat mendeteksi kecepatan tiap frame dari perpindahan objek. Respon perubahan data ketinggian dan kecepatan aliran dapat dideteksi tiap frame, namun dalam penampilan nilainya dilakukan tiap 5 detik. Beberapa hal yang mempengaruhi keberhasilan metode ini adalah gerakan objek, pencahayaan lingkungan, serta spesifikasi perangkat keras yang digunakan.*

**Kata kunci**—*pemrosesan video, aliran lahar dingin, optical flow, edge detection, region of interest.*

### Abstract

*Detection level and speed of cold lava flow for supporting early warning system using video processing is one system that uses computer vision system for the detection level and speed cold lava flow, which have not used speed detection sensors because they can be damaged by cold lava flow, and in the case of this is an early stage in the development of early warning systems towards.*

*The detection method is designed to detect the speed and height of the cold lava flow using Pyramidal Lucas Kanade optical flow, Sobel edge detection, thresholding, frame adder, hough transform, dilation and Region of Interest using. Programming using OpenCV 2.3.1 and using Visual Studio 2010. The programming language using C++.*

*The test results with some samples of the state shows that the method has a standard deviation value of 0.033 for the detection flow speed, and the standard deviation value of 0.035 for the detection of the flow height. Minimum and the maximum speed that can be detected depends on the type of camera used in the retrieval of video data, in this case with a camera system that is used to detect the speed of each frame of the object displacement. Response data*

*changes level and speed can be detected per frame, but the appearance of the value performed every 5 seconds. Some things that affect the success of this method is the movement of objects, ambient lighting, and hardware specifications used.*

**Keywords**—*video processing, cold lava flow, optical flow, edge detection, region of interest.*

## 1. PENDAHULUAN

Bencana alam dari letusan Gunung Merapi masih menyisakan material-material berat yang terkonsentrasi pada puncak gunung Merapi. Dari sumber informasi, bahwa dari sisa letusan November tahun 2010, material yang turun dari puncak Gunung Merapi adalah baru sekitar 30%. Sedangkan 70% sisanya masih terkumpul pada kawasan puncak kawah Gunung Merapi. Pada saat ini, sisa material hasil erupsi Merapi pada tahun 2010, masih sekitar 90 juta m<sup>2</sup> yang berada pada puncak gunung Merapi [1].

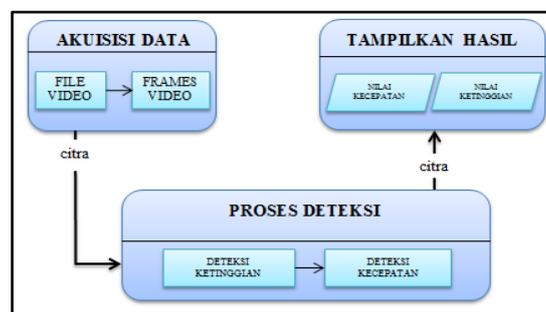
Sebuah jaringan sensor adalah sebuah infrastruktur yang terdiri dari elemen *sensing* (pengukuran), komputasi, dan elemen komunikasi yang memberikan kemampuan kepada administrator untuk melakukan instrumentasi, observasi, dan reaksi terhadap kejadian dan fenomena yang terjadi pada suatu lingkungan yang spesifik [2].

Pada penelitian ini akan dikembangkan suatu metode deteksi kecepatan dan ketinggian aliran lahar dingin berbasis pengolahan citra dan kemudian mengenali pola citra untuk mengolah data video rekaman pemantauan sehingga pendeteksian dapat dilakukan secara otomatis sehingga nantinya dapat digunakan sebagai suatu sistem untuk memperkirakan kapan datangnya aliran banjir lahar dingin pada aliran sungai bagian hilir, yang melewati tempat-tempat penting seperti pemukiman warga maupun situs-situs bersejarah, dan fasilitas umum yang vital.

## 2. METODE PENELITIAN

### 2.1. Rancangan Sistem Secara Keseluruhan

Rancangan sistem secara keseluruhan merupakan bagian dalam rancangan untuk perencanaan tahap-tahap yang diperlukan untuk membuat program pendeteksi kecepatan dan ketinggian aliran lahar dingin pendukung sistem peringatan dini. Program deteksi kecepatan dan ketinggian aliran lahar dingin dirancang untuk dapat memproses *frame* dari suatu file video aliran lahar dingin. Gambar 1 merupakan blok diagram rancangan algoritma sistem secara keseluruhan.



Gambar 1 Blok Diagram Rancangan Sistem Secara Keseluruhan

Pada Gambar 1 menggambarkan perancangan program secara keseluruhan, dimana tahap awal yang dilakukan adalah mengambil *file* video sampel yang akan dibaca nilai kecepatan dan ketinggian aliran air sungainya. Setelah itu dilakukan pembacaan *frame* dari file

video yang diambil untuk dilakukan proses deteksi pada proses selanjutnya. *Frame* yang diperoleh dari *file* video diproses satu per satu. .

Dalam rancangan algoritma ini menggunakan beberapa metode pengolahan citra, adapun beberapa metode yang digunakan dalam rancangan ini adalah sebagai berikut:

- Segmentasi objek dengan metode ROI (*Region Of Interest*) [3], metode ini digunakan untuk memfasilitasi penggambaran daerah yang penting dalam citra sungai yang akan dibaca kecepatan alirannya.
- *Frame Adder*, adalah metode yang digunakan untuk mengurangi frekuensi daerah aliran air sungai, dan menambah frekuensi daerah dinding sungai sehingga akan terlihat jelas perbedaan diantara kedua daerah tersebut[4].
- Operator Sobel[5], digunakan untuk mendeteksi garis tepi antara daerah aliran air sungai dengan dinding sungai.
- *Thresholding*[5], digunakan untuk proses mengubah citra berderajat keabuan menjadi citra biner atau hitam putih sehingga dapat diketahui perbedaan daerah mana yang termasuk *foreground* objek (dinding sungai) dan *background* dari citra yaitu aliran sungai secara jelas.
- *Dilation*[5], digunakan proses menambahkan piksel pada batasan dari objek dinding sungai sehingga nantinya citra dari dinding sungai akan terlihat hasilnya lebih besar ukurannya dibandingkan dengan citra aslinya.
- *Hough Transform*[5], digunakan untuk mendeteksi garis pada tepian dinding sungai sehingga akan didapat koordinat dari garis bibir sungai dan garis batas air sungai dengan dinding sungai.
- *Optical Flow Pyramidal Lucas Kanade*[6], digunakan untuk *tracking* pergerakan aliran air sungai sehingga dapat diperoleh perpindahan objek yang di tracking. Dalam hal ini menggunakan metode *Pyramidal Lucas Kanade*.

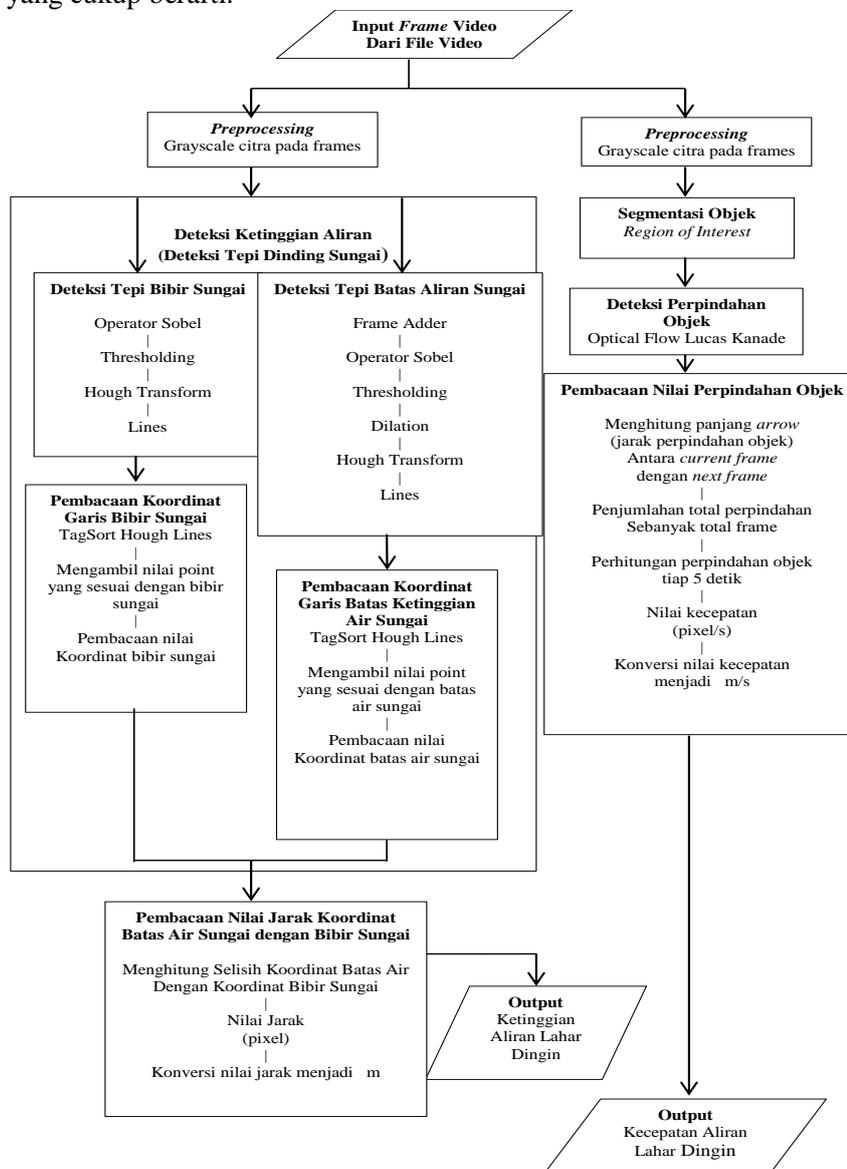
Diagram pada Gambar 3 menunjukkan algoritma sistem deteksi kecepatan dan ketinggian aliran lahar dingin secara keseluruhan. Algoritma pada awalnya menerima file video yang akan diproses, kemudian algoritma mengekstraksi *property* atau informasi file video berupa besaran ukuran resolusi, jumlah frame yang terdapat pada file, file video disini yang digunakan adalah file video berformat \*.avi. Kemudian terdapat proses yang terbagi dalam dua macam yaitu deteksi ketinggian air sungai dan kecepatan aliran air sungai. Pada deteksi ketinggian air, dilakukan beberapa proses algoritma yaitu yang pertama adalah preprocessing, merupakan proses dalam mengubah citra frame menjadi format *grayscale*. Setelah itu dilakukan proses deteksi dinding sungai, pada proses ini terbagi lagi menjadi dua bagian proses, yaitu proses deteksi tepi bibir sungai dan deteksi tepi batas air sungai menggunakan beberapa metode.

Proses deteksi bibir sungai menggunakan metode operator Sobel, *thresholding*, *hough transform*. Kemudian untuk deteksi batas air sungai ditambahkan metode *frame adder* sebelum operator Sobel dan metode *dilation* setelah proses *thresholding*. Untuk proses deteksi yang kedua adalah deteksi kecepatan aliran sungai, proses ini menggunakan metode *optical flowpyramidal Lucas Kanade* untuk mendeteksi perpindahan atau gerakan objek yang dibaca, karena metode *pyramidal Lucas Kanade* merupakan metode yang melakukan pembacaan tanpa menghilangkan gerakan yang kecil sehingga pada kasus ini tidak dilakukan pembatasan pembacaan perpindahan arah objek agar dapat terbaca semua gerakan objek yang bergerak. Objek yang dibaca adalah aliran lahar dingin yang bercampur batu kerikil serta lumpur, dalam hal ini *sample* yang digunakan adalah aliran sungai yang diambil di lokasi slokan Mataram sisi barat jalan Magelang yang ditunjukkan pada peta yang tertampil pada Gambar 2.



Gambar 2 Peta Lokasi Pengambilan Data *Sample Video*

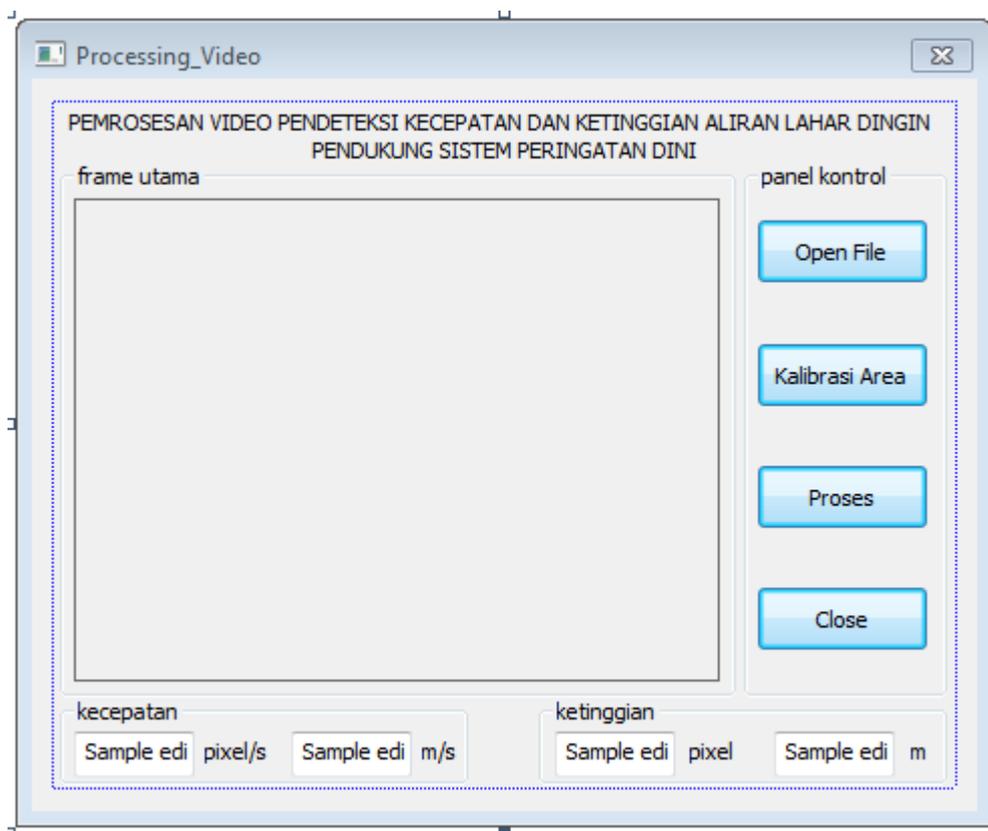
Alasan pengambilan data dilakukan slokan Mataram karena dinding sungai mirip dengan sabo tanggul penahan aliran lahar dingin, yang ada pada sungai-sungai yang berhulu di puncak gunung Merapi. Hal ini dilakukan karena kondisi sungai yang dilalui aliran lahar dingin berhulu di puncak gunung Merapi, dalam kondisi kering karena musim kemarau dan tidak ada aliran air yang cukup berarti.



Gambar 3 Diagram Algoritma Sistem Secara Keseluruhan

## 2.2. Implementasi

Implementasi metode pemrosesan video pendeteksi kecepatan dan ketinggian aliran lahar dingin menggunakan OpenCV 2.3.1 dan Visual Studio 2010 terdiri dari implementasi cara menghubungkan OpenCV 2.3.1 dengan Visual Studio 2010, kemudian implementasi cara mengakses *file* video, implementasi algoritma deteksi ketinggian aliran dengan deteksi bibir sungai dan deteksi batas aliran dengan dinding sungai, dan implementasi metode deteksi kecepatan aliran dan memindahkannya ke dalam bentuk GUI dari konsol Win32 ke aplikasi MFC, dan implementasi program *executable*, adapun tampilan ditunjukkan pada Gambar 4.



Gambar 4 Tampilan GUI Sistem Pemrosesan Video Pendeteksi Kecepatan Dan Ketinggian Aliran Lahar Dingin Pendukung Sistem Peringatan Dini.

## 2.3. Pengujian Metode Deteksi Kecepatan dan Ketinggian Aliran Lahar Dingin

Pengujian metode deteksi kecepatan dan ketinggian aliran lahar dingin pendukung sistem peringatan dini, dilakukan dengan cara mengaplikasikan metode deteksi kecepatan dan ketinggian aliran lahar dingin dengan beberapa sampel video yang telah direkam sebelumnya, beberapa *sample* video yang diambil dari dua posisi pengambilan. Dua posisi tersebut adalah posisi yang menghadap matahari yaitu video yang diberi nama (video\_2\_x) dan posisi yang membelakangi matahari yaitu video yang diberi nama (video\_1\_x). Hal ini dilakukan dengan terkait pencahayaan dari paparan sinar Matahari. Diperoleh dari kedua posisi tersebut, dalam metode deteksi dinding sungai diperlukan beberapa penyesuaian parameter. Untuk *file* video yang dicuplik selama 1 sekon tiap *file*-nya akan digunakan untuk kalibrasi sistem pendeteksi kecepatan dan ketinggian aliran lahar dingin pendukung sistem peringatan dini.

### 3. HASIL DAN PEMBAHASAN

#### 3.1. Hasil pengujian

Hasil pengujian meliputi tingkat keberhasilan deteksi kecepatan dan ketinggian aliran dari *frame* yang diproses, nilai deteksi kecepatan dan ketinggian aliran dari *frame* yang diproses, serta akurasi metode pemrosesan video pendeteksi kecepatan dan ketinggian aliran lahar dingin pendukung sistem peringatan dini terhadap sampel keadaan yang diujikan. Akurasi merupakan perbandingan kecepatan dan ketinggian aliran air pada objek pengambilan video yang berada di slokan Mataram dengan perhitungan yang dilakukan oleh program pemrosesan video pendeteksi kecepatan dan ketinggian aliran lahar dingin. Hasil pengujian terhadap sampel-sampel ditunjukkan pada Tabel 1 dan Tabel 2. Pada sampel video\_1\_6, video\_1\_7, dan video\_1\_8 di Tabel 1 terdapat nilai deteksi pembacaan ketinggian yang diberi warna kuning, hal ini karena nilai tersebut muncul akibat kesalahan deteksi garis batas aliran dengan dinding sungai saat proses pengujian berlangsung dengan ditunjukkannya nilai deteksi yang tidak wajar serta penggambaran garis batas aliran dengan dinding sungai yang tidak benar.

Tabel 1 Hasil Pengujian Sampel

No	Video Sampel	Kecepatan Aliran	Ketinggian Aliran	Kecepatan	Ketinggian
		Percobaan	Percobaan	Deteksi Program	Deteksi Pembacaan
		m/s	m	piksel/s	piksel
1	Video_1_1	1,73	0,55	1,37	58,67
2	Video_1_2	1,07	0,55	1,44	58,00
3	Video_1_3	1,07	0,55	1,14	58,00
4	Video_1_4	1,47	0,55	1,33	56,33
5	Video_1_5	1,33	0,55	1,15	55,33
6	Video_1_6	1,07	0,55	1,17	98,50
7	Video_1_7	1,07	0,55	1,80	109,50
8	Video_1_8	1,07	0,55	1,40	96,50
9	Video_2_1	2,63	0,55	2,35	53,80
10	Video_2_2	2,25	0,55	1,71	47,33
11	Video_2_3	2,38	0,55	3,05	49,75
12	Video_2_4	2,38	0,55	2,48	44,33
13	Video_2_5	1,50	0,55	2,05	51,33
14	Video_2_6	1,75	0,55	3,95	45,75
15	Video_2_7	2,38	0,55	1,20	50,50
16	Video_2_8	2,25	0,55	1,35	51,00

Tabel 2 Hasil Pengujian Sampel

N o	Nama Video	Pembacaan Kecepatan Aliran Menggunakan Flowmeter (m/s)	Pengukuran Ketinggian Aliran Menggunakan Alat Ukur Meter (m)	Pembacaan Kecepatan Aliran Menggunakan Program (piksel/s)	Pembacaan Ketinggian Aliran Menggunakan Program (piksel)
1	Video_3_1	0,41	0,55	5,23	43
2	Video_3_2	0,41	0,55	5,31	42
3	Video_3_3	0,42	0,55	5,34	43
4	Video_3_4	0,42	0,55	5,77	40
5	Video_3_5	0,44	0,55	5,97	40
6	Video_3_6	0,46	0,55	6,28	43
7	Video_3_7	0,46	0,55	6,51	42
8	Video_3_8	0,46	0,55	6,74	43
9	Video_3_9	0,47	0,55	6,85	42
10	Video_3_10	0,47	0,55	7,26	43
11	Video_3_11	0,47	0,55	8,03	43
12	Video_3_12	0,47	0,55	8,12	40
13	Video_3_13	0,47	0,55	8,28	44
14	Video_3_14	0,47	0,55	8,53	43
15	Video_3_15	0,47	0,55	8,63	43
16	Video_3_16	0,47	0,55	8,8	43
17	Video_3_17	0,47	0,55	9,3	43
18	Video_3_18	0,47	0,55	9,36	43
19	Video_3_19	0,48	0,55	9,37	43
20	Video_3_20	0,48	0,55	9,65	43
21	Video_3_21	0,48	0,55	9,65	43
22	Video_3_22	0,48	0,55	9,69	45
23	Video_3_23	0,48	0,55	9,94	41
24	Video_3_24	0,48	0,55	10,03	43
25	Video_3_25	0,48	0,55	10,68	43
26	Video_3_26	0,48	0,55	10,72	43
27	Video_3_27	0,49	0,55	11,04	45
28	Video_3_28	0,49	0,55	12,91	43
29	Video_3_29	0,5	0,55	13,07	43
30	Video_3_30	0,57	0,55	13,54	43

Pada Tabel 2 terdapat daftar nilai-nilai pembacaan program hasil deteksi kecepatan dan ketinggian aliran sungai pada *file* video sampel dengan satuan piksel/s untuk kecepatan dan piksel untuk ketinggian, namun untuk ketinggian di sini bukan ketinggian aliran namun jarak antara koordinat garis bibir sungai dengan garis koordinat batas aliran dengan dinding sungai sehingga pada program dilakukan perhitungan tinggi bibir sungai terhadap dasar sungai yang telah diukur yaitu sebesar 1,5 meter dikurangi jarak antara bibir sungai dengan batas aliran, sehingga diperoleh ketinggian aliran sungai pada video sampel. Hasil konversi atau kalibrasi

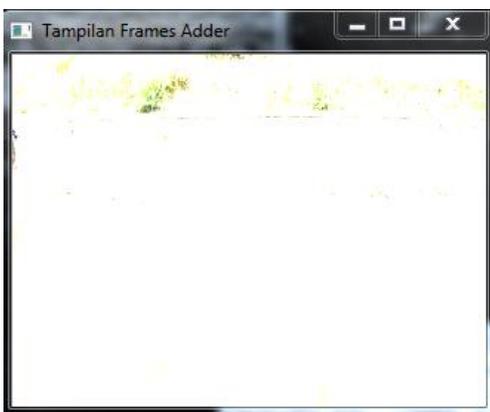
kecepatan aliran dari satuan piksel/s ke m/s dan ketinggian aliran dari satuan piksel ke meter pada video sampel ditunjukkan pada Tabel 3.

Tabel 3. Pendeteksian Kecepatan dan Ketinggian Aliran Telah Terkalibrasi

No	Nama Video	Pembacaan Kecepatan Aliran Menggunakan Flowmeter (m/s)	Pengukuran Ketinggian Aliran Menggunakan Alat Ukur Meter (m)	Pembacaan Kecepatan Aliran Menggunakan Program Setelah Dikalibrasi (m/s)	Pembacaan Ketinggian Aliran Menggunakan Program Konversi (m)
1	Video_3_1	0,41	0,55	0,44	0,68
2	Video_3_2	0,41	0,55	0,45	0,54
3	Video_3_3	0,42	0,55	0,45	0,54
4	Video_3_4	0,42	0,55	0,45	0,57
5	Video_3_5	0,44	0,55	0,45	0,54
6	Video_3_6	0,46	0,55	0,46	0,61
7	Video_3_7	0,46	0,55	0,46	0,61
8	Video_3_8	0,46	0,55	0,46	0,54
9	Video_3_9	0,47	0,55	0,47	0,57
10	Video_3_10	0,47	0,55	0,47	0,54
11	Video_3_11	0,47	0,55	0,47	0,57
12	Video_3_12	0,47	0,55	0,47	0,54
13	Video_3_13	0,47	0,55	0,48	0,61
14	Video_3_14	0,47	0,55	0,48	0,52
15	Video_3_15	0,47	0,55	0,48	0,54
16	Video_3_16	0,47	0,55	0,49	0,54
17	Video_3_17	0,47	0,55	0,49	0,54
18	Video_3_18	0,47	0,55	0,49	0,54
19	Video_3_19	0,48	0,55	0,5	0,54
20	Video_3_20	0,48	0,55	0,5	0,54
21	Video_3_21	0,48	0,55	0,5	0,54
22	Video_3_22	0,48	0,55	0,5	0,54
23	Video_3_23	0,48	0,55	0,52	0,59
24	Video_3_24	0,48	0,55	0,52	0,54
25	Video_3_25	0,48	0,55	0,53	0,54
26	Video_3_26	0,48	0,55	0,52	0,54
27	Video_3_27	0,49	0,55	0,52	0,5
28	Video_3_28	0,49	0,55	0,53	0,54
29	Video_3_29	0,5	0,55	0,55	0,54
30	Video_3_30	0,57	0,55	0,57	0,54

Secara keseluruhan, metode pemrosesan video pendeteksi kecepatan dan ketinggian aliran lahar dingin pendukung sistem peringatan dini, dengan sampel aliran air sungai sebagai media untuk kalibrasinya dapat mendeteksi dengan baik kecepatan dan ketinggian aliran air sungai dan ketinggiannya. Pada sampel yang dibuat beda posisi pengambilan sampelnya pada

Tabel 1, menunjukkan kesalahan deteksi pada ketinggian yang ditunjukkan pada sampel no 6, 7, dan 8. Kesalahan deteksi ini dikarenakan pada sampel video yang bernama video\_1\_1 hingga video\_1\_6 ini posisi pengambilan video pada objek, posisi kamera membelakangi cahaya matahari, gambar dari video hasil penangkapan jadi sangat terang hal ini terjadi karena kamera perekam yang digunakan memiliki teknologi auto *brighness*. Metode yang digunakan untuk deteksi ketinggian untuk video sampel video\_1\_x dengan video\_2\_x pada Tabel 1 mengalami perubahan pada seting parameter-parameternya, antara lain nilai iterasi untuk *dilation*, dan jumlah penambahan *frame adder*-nya. Untuk *frame adder* pada video\_1\_1 hingga video\_1\_8 hanya dilakukan penambahan sebanyak dua kali yaitu *frame* pertama dengan *frame* berikutnya, adapun hasil penambahan *frame* ini jika disesuaikan dengan parameter pada video\_2\_1 hingga video\_2\_8 maka video\_1\_x maka tampilannya ditunjukkan pada Gambar 5. Jika parameter *frame adder* digunakan untuk seri video\_2\_x maka hasilnya ditunjukkan pada Gambar 6.



Gambar 5 Proses Gagal  
Pada *Frame Adder*  
Untuk Seri Video\_1\_x



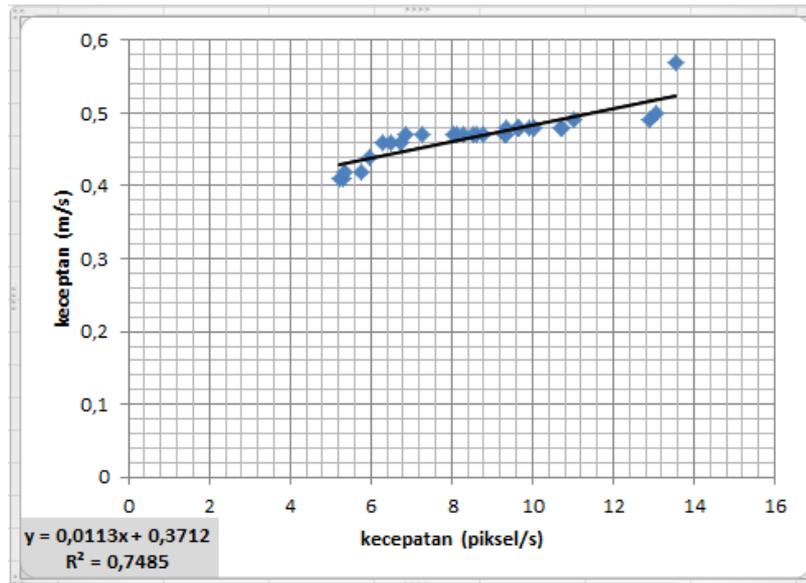
Gambar 6 Proses *Frame Adder* Yang Berhasil  
Pada Seri Video\_2\_x

Pada tampilan Gambar 5 menunjukkan bahwa proses *frame adder* gagal dengan penambahan *frame* sebanyak 8 kali, hal ini ditunjukkan pada gambar, citra yang dihasilkan terlalu terang dan daerah aliran sungai dengan dinding sungai tidak terlihat sehingga deteksi batas aliran sungai tidak dapat terdeteksi, sehingga dilakukan pengurangan parameter untuk *frame adder* hanya dilakukan 2 kali, dan hasilnya ditunjukkan pada Gambar 7.



Gambar 7 Tampilan *Frame Adder* Untuk Video\_1\_x Sebanyak 2 kali

Pada Video sampel Tabel 2 menunjukkan hasil pembacaan nilai kecepatan dan ketinggian aliran video sampel, dengan satuan piksel/s untuk kecepataannya dan satuan piksel untuk ketinggiannya. Kemudian untuk konversi nilai piksel/s ke m/s dan piksel ke meter, maka dilakukan konversi menggunakan metode kalibrasi regresi linier untuk nilai kecepataannya dengan persamaan  $y = mx + a$ , dimana  $y$  adalah hasil yang diinginkan dan  $x$  adalah nilai dari kecepatan dengan satuan piksel/s. Dengan membuat grafik antara kecepatan piksel/s sebagai sumbu  $x$  dan kecepatan m/s (*flow meter*) sebagai sumbu  $y$  maka dapat diperoleh nilai  $m$  dan  $a$  dari persamaan  $y = mx + a$ , adapun gambar grafik untuk kalibrasi ini ditunjukkan pada Gambar 8.



Gambar 8 Grafik Kecepatan (piksel/s) dengan Kecepatan (m/s *flow meter*)

Dari grafik yang ditunjukkan pada Gambar 9. diperoleh persamaan;

$$y = 0,0113x + 0,3712 \quad (1)$$

dengan

$$R^2 = 0,7485 \quad (2)$$

Persamaan (2) ini diimplementasikan dalam program untuk mengkonversi nilai kecepatan yang terbaca piksel/s menjadi m/s. Adapun hasil pembacaan data *file* video ditunjukkan pada Tabel 3. Video-video sampel pada Tabel 3 adalah video sampel yang berdurasi 1 detik. Untuk konversi ketinggian adalah dengan mengambil rata-rata dari total data pembacaan piksel kemudian digunakan sebagai pembagi dan kemudian dikalikan nilai 0,95 sebagai nilai ketinggian pembacaan menggunakan alat ukur meter dengan satuan meter. Persebaran data pada Tabel 3 jika dicari nilai standar deviasinya untuk kecepatan dan ketinggian aliran maka diperoleh nilai untuk standar deviasi data deteksi kecepatan oleh program 0,033 dan nilai untuk standar deviasi data deteksi ketinggian oleh program sebesar 0,035 dengan akurasi sebesar 74,9% untuk deteksi kecepataannya, sehingga dapat dikatakan program ini melakukan deteksi kecepatan dan ketinggian aliran memiliki presisi yang cukup tinggi.

Adapun untuk batas terdeteksinya kecepatan masih dapat terdeteksi adalah selama pergerakan benda atau objek tidak melebihi batas kecepatan kamera perekam maka kecepatan tersebut dapat terdeteksi karena pada dasarnya metode *optical flow* adalah metode yang melakukan *tracking* pergerakan objek tiap *frame*-nya. Dengan demikian metode ini dapat mengatasi masalah pembacaan kecepatan dan ketinggian aliran lahar dingin karena dengan metode ini maka dapat digunakannya pengolahan pola pada citra menggunakan kamera yang

peletaknya dapat di tempat aman dari terjangan aliran lahar dingin, artinya dapat digunakan untuk pendukung sistem peringatan dini dengan penelitian lebih lanjut. Sedangkan untuk mengimplementasikannya maka diperlukannya seting ulang mengenai jarak kamera dengan objek kemudian sudut kamera serta seting parameter-parameter yang disesuaikan dengan keadaan lingkungan pengamatan, dalam hal ini kamera perekam untuk sampel video yang diambil memiliki jarak 1 meter dari bibir sungai, dengan ketinggian 1,5 meter dan sudut kamera sebesar 65 derajat terhadap sumbu y. metode ini sangat baik untuk objek deteksi yang tingkat kecerahannya tidak terlalu tinggi, tingkat kecerahan yang tinggi ini dikarenakan paparan sinar matahari yang memantul pada permukaan air maupun dinding sungai, sehingga menyulitkan untuk pendeteksian ketinggian aliran.

Dalam metode ini menggunakan sampel aliran air sungai biasa, bukan aliran lahar dingin karena saat pembuatan sistem ini dan saat kalibrasi tidak adanya data sampel yang dapat digunakan ujicoba dan kalibrasi, karena dibutuhkan kamera yang statis dan tersetting untuk pengambilan video, sehingga digunakan sampel aliran air sungai. Adapun metode ini nantinya tetap dapat digunakan untuk pembacaan aliran lahar dingin, karena dalam deteksi kecepatan menggunakan segmentasi objek berupa gerakan benda atau aliran air yang mengalir, sehingga jika nanti digunakan untuk aliran lahar dingin maka metode ini dapat diterapkan.

Video pada *frame* saat proses deteksi gerakannya cukup lambat karena di dalam proses terdapat operasi-operasi seperti *hough transform*, operator sobel, dan *dilation*. Namun demikian hal ini tidak terlalu berpengaruh pada proses deteksi karena pembacaan dilakukan tiap *frame* dari video, namun penampilannya dilakukan tiap lima detik.

#### 4. KESIMPULAN

1. Metode ini dapat membaca nilai kecepatan dan ketinggian dengan presisi tinggi, dengan dilihat standar deviasi kecepatan sebesar 0,033 dan standar deviasi untuk ketinggian sebesar 0,035. Sedangkan untuk akurasi deteksi kecepatan aliran terhadap sampel video sebesar 74,9
2. Metode ini dapat mendeteksi kecepatan dan ketinggian aliran lebih baik jika sisi atau daerah yang diamati tidak memiliki kecerahan yang rendah.
3. Metode *Optikal FlowPyramidal* Lucas Kanade dapat digunakan untuk pembacaan gerakan objek yang terbawa aliran maupun gerakan arus yang terdapat pada aliran sungai.
4. Metode ini dapat membaca kecepatan aliran dengan membatasi wilayah deteksi menggunakan *Region of Interest*.
5. Akurasi keberhasilan dipengaruhi oleh gerakan objek, posisi dan arah kamera saat pengambilan video, tingkat serta faktor pencahayaan.
6. Batas kemampuan deteksi kecepatan dan ketinggian aliran adalah sesuai dengan fitur dari kamera yang digunakan, selama kamera masih dapat menangkap gambar video dengan baik maka metode ini dapat melakukan pembacaan.
7. Sistem pemrosesan video pendeteksi kecepatan dan ketinggian aliran lahar dingin pendukung sistem peringatan dini dengan dapat berjalan dengan baik.
8. Sistem belum teruji untuk implementasi sehingga perlu dilakukan penelitian lebih lanjut.

#### 5. SARAN

Beberapa saran dari penulis untuk pengembangan metode deteksi wajah *real time* lebih lanjut sebagai berikut:

1. Memilih letak posisi kamera yang baik, karena sangat menentukan hasil video yang akan diamati, hal ini terkait pencahayaan dan gangguan-gangguan yang dapat mengganggu proses deteksi.
2. Kalibrasi kecepatan hendaknya menggunakan kalibrator alat ukur kecepatan fluida cairan yang dapat mengatasi aliran turbulen sehingga diperoleh akurasi yang tinggi sehingga hasil kalibrasi dapat lebih baik.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada UGM yang telah memberi dukungan finansial terhadap penelitian ini.

## DAFTAR PUSTAKA

- [1] Subandrio, 2011. februari-maret-puncak-lahar-dingin-2012, subandrio.html. diakses maret 2012 jam 19.31 WIB.
- [2] K. Sohraby, D. Minoli, T. Znati, “*Wireless Sensor Network: Technology, Protocol, and Applications*”, Wiley-interscience, 2007.
- [3] Nasrudin, 2009, *OpenCV Region of Interest (ROI)* [http://nashruddin.com/OpenCV\\_Region\\_of\\_Interest\\_%28ROI%29](http://nashruddin.com/OpenCV_Region_of_Interest_%28ROI%29) diakses pada tanggal 1 Oktober 2012 jam 21.00 WIB.
- [4] Iwahashi, M., Udomsiri, S., 2007. *Water Level Detection from Video with FIR Filtering*, in: *Computer Communications and Networks*, 2007. ICCCN 2007. Proceedings of 16th International Conference On. pp. 826–831.
- [5] Ahmad, U., 2005, *Pengolahan Citra Digital & Teknik Pemogramannya*, Penerbit Graha Ilmu, Yogyakarta.
- [6] Lin, W., 2011. *Video Surveillance*. InTech Janeza Trdine 9 51000 Rijeka, Croatia.