

Sistem Pemantauan Volume Timbulan Sampah berbasis *Internet of Things* di Tempat Penampungan Sementara Kota Bogor

Internet of Things based Volume Monitoring System for Waste Disposal in Bogor City

SRI WAHJUNI¹, WULANDARI^{1*}, RIZQI ALIFAHASNI ZAKIAH¹

Abstrak

Sampah masih menjadi salah satu masalah yang sulit untuk ditangani oleh pemerintah Kota Bogor. Dengan jumlah penduduk mencapai angka 1.08 juta jiwa, jumlah sampah yang dihasilkan per harinya mencapai 2,532.5 liter. Jumlah sampah yang tergolong banyak ini hanya diakomodasi oleh jumlah armada angkutan sampah yang terbatas. Hal ini mengakibatkan sampah di Tempat Penampungan Sementara (TPS) menumpuk sehingga mencemari lingkungan. Oleh karena itu, diperlukan sebuah sistem pemantauan jarak jauh untuk mengetahui volume timbulan sampah. Penelitian ini membangun sebuah prototype untuk mengetahui volume timbulan sampah secara *real-time* dan memberikan notifikasi untuk pengambilan sampah di luar jadwal regular, yaitu ketika volume timbulan sampah mendekati volume maksimal bak sampah. Metode yang digunakan dalam penelitian adalah *frame difference algorithm* dan *perspective transformation*. Selain itu, penelitian ini juga menerapkan konsep *Internet of Things* dalam aplikasi berbasis *mobile* sebagai sistem pemantauan dan notifikasi. Nilai *error* atau RMSE antara tinggi aktual volume timbulan sampah dan tinggi timbulan volume sampah hasil pengolahan citra yang didapat adalah 0.2060 cm dan akurasi sebesar 97.93 persen.

Kata Kunci: *frame difference algorithm*, *internet of things*, pengangkutan sampah, *perspective transformation*, *smart city*

Abstract

Waste management is one of ongoing problem that difficult to handle by the government of Bogor City. With a population reaching 1.08 million, the amount of waste produced per day reaches 2 532.5 liters. This large amount of waste only accommodated by limited number of garbage transport fleets. This accumulates the trash in the waste disposal (TPS) and pollutes the environment. Therefore, a remote monitoring system is needed to determine the volume of waste generation. This study will build a prototype to determine the volume of waste generation in real-time and detect the volume of waste and provide notifications for garbage collection outside the regular schedule. When the volume of waste close to the maximum volume of garbage bins. The method used in this research is frame difference algorithm and perspective transformation. In addition, this study also applies the Internet of Things concept in mobile-based application as a monitoring and notification system. The error value or RMSE between the actual height of the waste volume and the height of the waste volume measured by the image processing is 0.2060 with accuracy 97.93 percent.

Keywords: *frame difference algorithm*, *garbage transportation*, *internet of things*, *perspective transformation*, *smart city*

¹Departemen Ilmu Komputer Jl Meranti Wing 20 Level 5 Kampus IPB Darmaga 16680;

*Penulis korespondensi: wulandari.ilkom@apps.ipb.ac.id

PENDAHULUAN

Pengelolaan sampah yang belum terpadu terutama masalah volume timbulan sampah menjadi salah satu dari tiga isu prioritas Kota Bogor yang harus segera ditangani (DLH 2017). Meningkatnya volume timbulan sampah saat ini disebabkan oleh perubahan tingkat populasi dan gaya hidup, yaitu semakin maju dan sejahtera kehidupan seseorang, maka semakin tinggi juga jumlah sampah yang dihasilkan (El-Haggar 2007). Data Badan Pusat Statistik Kota Bogor tahun 2017 menunjukkan bahwa terdapat 1.08 juta jiwa di Kota Bogor sehingga kota ini dapat dikategorikan sebagai kota besar (BPS 2018). Jumlah sampah yang dihasilkan penduduk Kota Bogor pada tahun 2014 adalah 2,532.5 meter kubik per hari dan hanya 1,811.5 meter kubik sampah yang terangkut (DLH 2015).

Sampah perkotaan yang dapat diangkut dari Tempat Penampungan Sementara (TPS) ke Tempat Pemrosesan Akhir (TPA) diperkirakan hanya 40 hingga 50 persen dari total sampah (Damanhuri 2005). Sampah yang telah terkumpul di TPS diangkut ke TPA menggunakan armada angkutan sampah dari Dinas Kebersihan dan Pertamanan (DKP) Kota Bogor. Jadwal pengangkutan sampah tidak sama pada setiap jalur pengangkutan dan jadwal disesuaikan dengan jumlah timbulan sampah di setiap titik angkut (Bappeda 2014; Rusmaya *et al.* 2019). Salah satu contoh masalah berkaitan dengan volume timbulan sampah adalah ketika diselenggarakan aktivitas atau acara besar di tingkat kecamatan (Samsuri dan Maulana 2019). Akibatnya, terjadi peningkatan jumlah sampah secara drastis dan volume timbulan sampah ini akan terus bertambah hingga pengangkutan sampah berikutnya tiba. Dampak negatif yang terjadi diantaranya adalah pencemaran, ketidaknyamanan, dan merusak keindahan lingkungan.

Kota Bogor adalah salah satu kota terpilih di Indonesia untuk menjalankan program kerja Gerakan Menuju 100 *Smart City* Indonesia (Diskominfo 2017). Cohen Boyd mencetuskan *Cohen's smart cities wheel* pada tahun 2012 yaitu diagram yang mendefinisikan *smart city* dengan enam pilar utama, salah satunya adalah *smart environment* (Fastcompany 2012). Salah satu sasaran pemerintah Kota Bogor untuk *smart environment* adalah meningkatkan tata kelola persampahan berbasis masyarakat. Indikator sasaran ini adalah keberhasilan meningkatkan volume sampah yang terangkut ke TPA (Diskominfo 2017). Memonitor volume sampah dari jarak jauh dan memberikan notifikasi merupakan salah satu cara untuk membantu upaya peningkatan volume sampah yang terangkut ke TPA. Dengan mengetahui volume timbulan sampah di TPS secara *real-time*, pemerintah dapat membuat daftar prioritas lokasi bak sampah yang perlu dibersihkan sehingga jadwal pengangkutan sampah lebih fleksibel dan efektif. Volume timbulan sampah dapat diketahui dengan memanfaatkan teknik pengolahan citra yaitu *background subtraction*.

Frame difference algorithm merupakan salah satu metode dari *background subtraction*. Metode ini digunakan untuk memisahkan *background* dan *foreground* dari citra yang diinginkan dengan cara mengurangi dua citra berurutan. Jika nilai piksel dari hasil pengurangan lebih besar dari *threshold* yang ditentukan, maka akan dipertimbangkan sebagai *foreground* (Singla 2014). Penelitian terkait deteksi citra menggunakan *background subtraction* telah banyak dilakukan salah satunya yaitu pengembangan sistem tertanam waktu nyata untuk menghitung jumlah kendaraan di jalan raya oleh Muqaddami (2017). Citra *foreground* yang telah didapatkan ditransformasi dari 3D objek menjadi citra 2D dengan menggunakan fungsi *perspective transformation*. Citra yang telah ditransformasi digunakan untuk mengukur tinggi dan volume timbulan sampah. Pengolahan citra menggunakan *frame difference algorithm* dapat menjadi salah satu alternatif yang dapat digunakan untuk mengatasi masalah sampah di Kota Bogor. Algoritme ini dapat digunakan untuk menghitung volume timbulan sampah di TPS. Dengan

bantuan teknologi, DKP dapat memantau volume timbulan sampah dari jauh dan mengetahui prioritas titik lokasi TPS yang harus diangkut.

Internet of Things (IoT) adalah sebuah kemampuan benda mati untuk berinteraksi dengan objek lain, lingkungan maupun dengan peralatan komputasi cerdas lainnya melalui koneksi jaringan. Salah satu contoh IoT adalah sebuah perangkat yang digunakan untuk mengontrol dan memantau lingkungan (Uckelmann *et al.* 2011). Konsep ini dapat diterapkan untuk mengatasi masalah penumpukan sampah di TPS kecamatan Kota Bogor. Dengan IoT, volume timbulan bak sampah dapat dipantau dari jarak jauh. Keuntungan lain dari penggunaan IoT adalah pengukuran dan pengamatan dilakukan secara *real-time*. Protokol yang banyak digunakan pada penelitian berbasis IoT adalah protokol *representational state transfer* (REST). Penelitian dengan konsep IoT dan REST sebagai protokol telah banyak dilakukan salah satunya yaitu penerapan pada sistem pengendali otomatis irigasi oleh Waladi (2017). Penelitian ini bertujuan untuk mengontrol sistem irigasi dengan menggunakan sistem keputusan *fuzzy* yang diterapkan pada *single board komputer* (SBC) yang terhubung pada tiga buah sensor yaitu *kelembaban* dan suhu (DHT 22) dan sensor ultrasonic (HC-SR04).

REST adalah satu jenis *web service* yang menerapkan konsep perpindahan antar *state* yang dianalogikan seperti jika *browser* meminta suatu halaman web, maka server akan mengirimkan *state* halaman web yang sekarang ke *browser* (Hariyanto *et al.* 2013). REST bekerja dengan bernavigasi melalui link *Hypertext Transfer Protokol* (HTTP) dengan perintah dasar GET, POST, PUT, dan DELETE. Keuntungan penggunaan IoT dengan protokol REST adalah mudah untuk diintegrasikan pada layanan *client* yang telah tersedia dan relatif ringan.

Penelitian ini bertujuan untuk membangun sebuah sistem yang mengimplementasikan *frame difference algorithm* dan *perspective transformation* untuk mengetahui volume timbulan sampah pada bak sampah. Nilai volume dan level tinggi sampah dikirim ke server menggunakan protokol REST untuk digunakan dalam sistem pemantauan dan pembangkitan notifikasi.

METODE

Tahapan penelitian yang dilakukan pada penelitian ini adalah identifikasi masalah, studi literatur, perancangan arsitektur sistem, implementasi, integrasi, dan pengujian dan evaluasi.

Identifikasi Masalah

Tahapan identifikasi masalah dilakukan untuk mengetahui masalah-masalah yang terjadi di lapangan. Kebutuhan sistem pengambilan citra ini menggunakan modul kamera, perangkat pengolahan data berupa Raspberry Pi, bahasa yang digunakan adalah python.

Studi Literatur

Studi literatur dilakukan untuk mengumpulkan informasi yang berkaitan dengan penelitian serta untuk memperkuat ilmu dasar mengenai IoT dan protokol jaringan sensor. Tahapan ini juga dilakukan sebagai tahap observasi terhadap penelitian-penelitian ataupun sistem yang sudah ada sebelumnya untuk dikembangkan dalam penelitian, terutama dalam bidang pengolahan citra dengan menggunakan teknik *frame difference algorithm* dan *perspective transformation*.

Perancangan Arsitektur Sistem

Perancangan ini terdiri atas dua bagian yaitu perancangan *hardware* dan perancangan *software*. Mencakup proses membangun sistem yaitu sistem pengolahan citra untuk

mendapatkan nilai volume, perancangan posisi sensor dan Raspberry Pi dalam bak sampah, dan sistem yang digunakan untuk pemantauan.

Implementasi

Perakitan alat dilakukan pada tahapan ini yaitu peletakkan posisi sensor dan Raspberry Pi di miniatur bak sampah. Algoritme yang telah dirancang pada tahapan sebelumnya diimplementasikan menjadi sebuah program pada tahapan ini.

Integrasi

Tahapan ini merupakan tahap pengintegrasian antara sensor dan Raspberry Pi yang telah dirakit dengan aplikasi berbasis *mobile*. Perlu diperhatikan juga kecocokan antara program atau bahasa yang digunakan dengan alat.

Pengujian dan Evaluasi

Tahapan ini digunakan untuk menguji dan mengevaluasi sistem yang telah dibuat dimulai dari pengambilan citra menggunakan *node* sensor, mengolah citra, mencari volume timbulan sampah, mengirim data ke server, dan membangkitkan notifikasi.

1 Uji Perhitungan Tinggi dan Pengaruh Intensitas Cahaya

Uji perhitungan volume dilakukan untuk melihat seberapa baik dan akurat algoritme dan instrumen yang digunakan. Pengujian dilakukan dengan cara membandingkan hasil pengukuran menggunakan alat ukur standar dengan hasil pengukuran menggunakan instrumen yang telah terpasang kode program yang digunakan. Dengan melakukan uji perhitungan volume, penyimpangan antara harga benar dan harga yang ditunjukkan oleh instrumen dapat diketahui.

Penyimpangan data yang mungkin terjadi sebagai evaluasi dari sistem diukur dengan menggunakan *root mean square error* (RMSE). RMSE disebut juga sebagai simpangan baku nilai hitung dengan nilai sebenarnya. Rumus dari fungsi RMSE dapat dilihat pada Persamaan 1.

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n e_i^2} \quad (1)$$

Keterangan:

e = error antara nilai hitung dan nilai sebenarnya.

Mean Absolute Percentage Error (MAPE) dihitung untuk mengetahui persentase galat dari data yang didapat. Hasil dari MAPE dikali dengan 100 untuk mendapatkan nilai dalam bentuk persentase. Sedangkan untuk mendapatkan akurasi data, hitung $100\% - \text{MAPE}$. Semakin kecil nilai MAPE, maka data semakin bagus. Rumus untuk fungsi MAPE dapat dilihat pada Persamaan 2.

$$\text{MAPE} = \frac{\sum_{t=1}^n \frac{|At - Ft|}{At}}{n} \times 100 \quad (2)$$

Keterangan:

At = nilai aktual

Ft = nilai prediksi

Selain itu, pengaruh intensitas cahaya terhadap akurasi perhitungan tinggi timbulan sampah juga diuji dengan membandingkan hasil antara penangkapan citra di luar ruangan, di dalam ruangan, dan di tempat gelap atau minim cahaya.

2 Uji Fungsional Protokol REST

Pengujian fungsional dilakukan untuk mengetahui apakah sistem yang dibuat pada penelitian ini dapat berintegrasi dengan server dan aplikasi berbasis *mobile* hasil penelitian Waladi (2017). Dalam penelitiannya, terdapat 10 *request* utama yang dapat dilihat pada Tabel 1. Penelitian ini hanya mengadaptasi tujuh *request* dari 10 *request* utama. Metode yang digunakan untuk pengujian ini adalah metode *black-box* dengan menggunakan *software* Postman.

3 Uji Sistem Pemantauan dan Notifikasi

Uji sistem pemantauan adalah tahap menguji apakah proses pemantauan dari aplikasi berbasis *mobile* berjalan dengan baik, yaitu visualisasi data volume yang telah tersimpan di dalam server dan pembangkitan notifikasi. Pengujian *test case* juga dilakukan untuk menguji pembangkitan notifikasi sesuai dengan *threshold* yang telah ditentukan.

Tabel 1 API REST Penelitian Waladi (2017)

No	Method	URL	Hasil respons
1	GET	http://[hostname]:3000/api/devices/:id	Mendapatkan semua device berdasarkan id <i>user</i>
2	POST	http://[hostname]:3000/api/device/create	Membuat <i>device</i> baru
3	GET	http://[hostname]:3000//api/sensornode/:id	Mendapatkan <i>node</i> sensor berdasarkan id <i>device</i>
4	POST	http://[hostname]:3000//api/sensornode/create	Membuat <i>sensornode</i> baru
5	GET	http://[hostname]:3000/api/dataset/:id	Mendapatkan semua data berdasarkan id <i>node</i> sensor
6	POST	http://[hostname]:3000/api/dataset/create	<i>Push</i> data dari Raspberry Pi ke <i>database</i>
7	POST	http://[hostname]:3000/api/notification/send	Mengirim notifikasi ke satu <i>mobile device</i>
8	POST	http://[hostname]:3000/api/prediction/gets	Mendapatkan data hasil prediksi nilai sensor
9	POST	http://[hostname]:3000/api/control	Menyalakan / mematikan aktuator
10	PUT	http://[hostname]:3000//api/sensornode/:id/update	Pembaharuan informasi dari satu <i>node</i> sensor

HASIL DAN PEMBAHASAN

Identifikasi Masalah

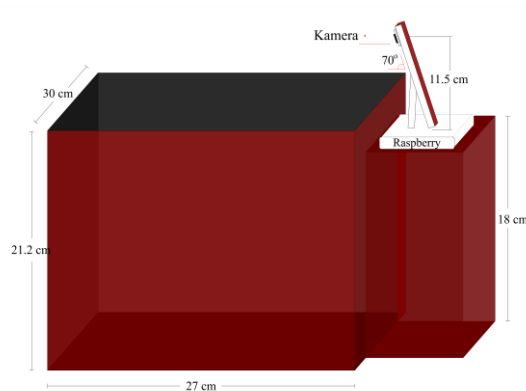
Waste management atau pengelolaan sampah di Kota Bogor perlu dikelola dengan baik untuk mengurangi dampak negatif baik terhadap lingkungan, kesehatan, maupun keindahan. *Waste management* merupakan kegiatan yang mencakup pengumpulan, pengangkutan, daur ulang, dan pembuangan dari materiil sampah. Pengangkutan dilakukan dengan cara manual, yaitu sampah dari berbagai sumber (rumah tangga, pertanian, industri, perdagangan, dan perkantoran) dikumpulkan dengan menggunakan gerobak sampah menuju TPS. Setelah itu dilakukan tahap akhir, yaitu pengangkutan dengan armada angkutan sampah yang disediakan DKP untuk dibuang ke TPA.

Menurut data Badan Pusat Statistik (BPS) jumlah penduduk kota Bogor yang tersebar di enam kecamatan mencapai 1 081 009 jiwa dengan penyebaran yang tidak merata, sehingga jumlah tabungan sampah yang dihasilkan setiap TPS kecamatan berbeda. Kapasitas angkut dan jumlah armada angkutan sampah tidak sebanding dengan jumlah sampah yang harus ditransportasi dari TPS menuju TPA. Hal ini dapat menimbulkan masalah ketika TPS memiliki timbunan sampah berlebih sedangkan pengangkutan sampah selanjutnya masih terhitung lama. Contoh situasi tersebut adalah ketika suatu daerah melaksanakan sebuah acara maupun aktivitas besar yang menghasilkan residu sampah melebihi biasanya. Sampah yang melebihi kapasitas penampungan tersebut dapat berceceran hingga keluar bak sampah dan mencemari lingkungan, sehingga berdampak buruk bagi kesehatan dan merusak keindahan.

Berdasarkan permasalahan tersebut, salah satu solusi yang dapat dilakukan adalah dengan memanfaatkan Raspberry Pi sebagai alat untuk mencari volume timbunan sampah. Penerapan konsep IoT juga dapat dilakukan untuk memudahkan pihak DKP memantau perkembangan volume sampah dari jarak jauh.

Perancangan Arsitektur Sistem

Pengambilan gambar tidak dilakukan langsung di TPS kecamatan Kota Bogor, melainkan dengan menggunakan miniatur bak sampah berukuran panjang 30 cm, lebar 27 cm, dan tinggi 21.2 cm yang dibuat sendiri menggunakan bahan kardus. Kamera diletakkan menunduk ke arah bagian dalam bak sampah agar jangkauan tangkapan citra luas, dari atas bak sampah hingga dasar bak sampah. Sudut kemiringan kamera adalah 70 derajat. Rancangan arsitektur miniatur bak sampah dapat dilihat pada Gambar 1.



Gambar 1 Perancangan arsitektur miniatur bak sampah (tampak depan)

Perangkat Keras

Penelitian ini akan membuat *prototype* yang menggunakan modul kamera untuk pengambilan citra dan Raspberry Pi untuk perangkat pengolahan. Alat dan bahan yang diperlukan selama penelitian adalah sebagai berikut.

- Raspberry Pi 3 model B dengan sistem operasi Raspbian Jessie untuk melakukan pengolahan citra dan pengiriman data menuju server
- Modul kamera Raspberry Pi Rev 1.3 dengan spesifikasi yaitu resolusi sebesar 5 megapiksel dan kualitas citra 2592 x 1944 piksel
- Satu buah Micro SD Card 16 GB
- Acer Espire E 14 dengan spesifikasi *processor* Core i5 dan RAM 4GB digunakan sebagai lingkungan pengembangan

Pengambilan citra di TPS kecamatan Kota Bogor pada pengaplikasiannya dilakukan setiap 6 jam yaitu pukul 6 pagi, pukul 12 siang, pukul 6 sore, dan pukul 12 malam. Sedangkan pada penelitian ini dilakukan menggunakan miniatur bak sampah dan perencanaan pengambilan citra dilakukan setiap 60 detik dalam sepuluh menit. Citra yang ditangkap akan disimpan di dalam *memory* Raspberry Pi. Untuk mengetahui kebutuhan *memory* dalam penelitian ini, dilakukan perhitungan sebagai berikut:

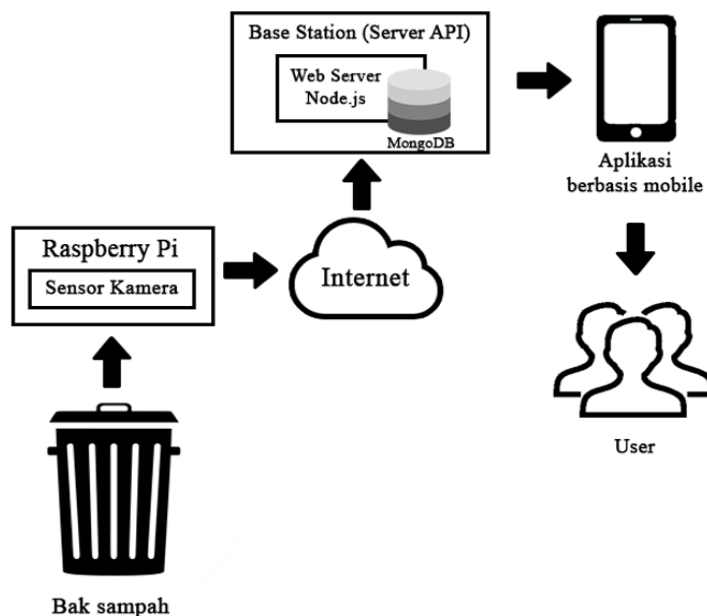
- Satu citra memiliki resolusi 670 x 600 piksel, setiap citra yang diambil diasumsikan memiliki *size* 300 kB
- Dalam satu hari dilakukan pengambilan citra sebanyak 4 kali sehingga $4 \times 300 = 1200$ kB
- *Memory* SD Card yang digunakan dalam penelitian ini adalah 16 GB. *Memory* yang telah terpakai untuk *operating system* adalah 4.3 GB dan OpenCV 2 GB. *Memory*

yang tersisa untuk penyimpanan citra adalah 9.7 GB. *Memory* SD Card dapat menampung hingga 32 333 citra atau setara dengan 8 083 hari pengambilan citra.

Skema perancangan *prototype* dapat dilihat pada Gambar 2. Modul kamera menangkap citra untuk selanjutnya dilakukan pengolahan perhitungan volume dan level oleh Raspberry Pi. Data volume dan level yang didapat dikirim ke server menggunakan metode POST dan disimpan di dalam *database*. Data dapat dipantau melalui aplikasi berbasis *mobile*. Pembangkitan notifikasi terjadi apabila nilai level yang didapat adalah 3 atau 4. Hal ini menunjukkan bahwa bak sampah harus segera dibersihkan.

Perangkat Lunak

Perangkat lunak yang dibutuhkan selama pengembangan penelitian adalah Python 3.6.6 dengan *library* bantuan OpenCV 3.4.3 sebagai bahasa pemrograman untuk pengolahan citra, MongoDB 3.6 sebagai DBMS, dan Lux Light Meter Free versi 015.2018.11.07 untuk mengukur intensitas cahaya.



Gambar 2 Arsitektur IoT sistem pemantauan volume timbulan sampah.

Pengolahan citra yang dilakukan dalam penelitian ini adalah dengan penerapan teknik *frame difference algorithm* dan *perspective transformation*. *Background Subtraction* digunakan untuk memisahkan *foreground object* dari latar belakang pada citra. *Frame difference algorithm* adalah salah satu metode *background subtraction* yang paling banyak digunakan karena memiliki kecepatan proses yang cepat dan juga dapat diimplementasikan dengan mudah. Algoritme ini bekerja dengan cara mengurangi dua buah citra masukan dengan $f_k[i][j]$ merupakan piksel dari *frame* pertama untuk dijadikan *background* dan $f_{k-1}[i][j]$ adalah piksel *frame* berikutnya. Untuk mendapatkan nilai *foreground*, dilakukan pengurangan nilai citra f_{k-1} dengan nilai citra f_k sehingga menghasilkan citra $D_k[i][j]$. Rumus dari $D[i][j]$ dapat dilihat pada Persamaan 3.

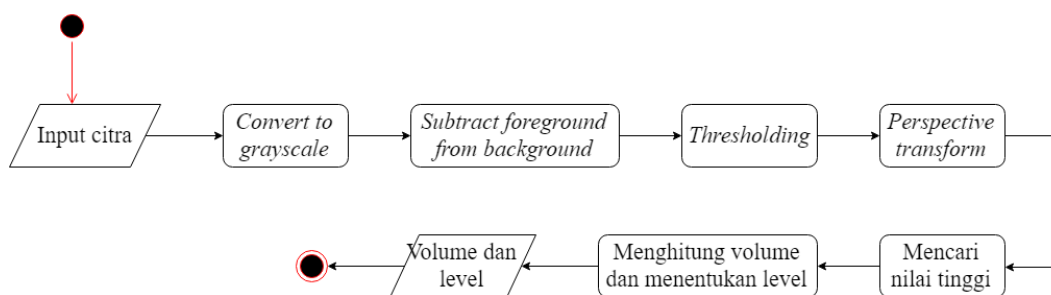
$$D_k[i][j] = \begin{cases} 1, & |f_{k-1}[i][j] - f_k[i][j]| \geq T \\ 0, & |f_{k-1}[i][j] - f_k[i][j]| < T \end{cases} \quad (3)$$

Gaussian blur dan *ss binarization* dilakukan untuk menghilangkan *noise* dan mengubah citra menjadi citra biner. Jika nilai piksel melebihi atau sama dengan nilai *threshold*, maka nilai piksel $D_k[i][j]$ adalah 255 atau putih. Jika nilai piksel kurang dari *threshold*, nilai piksel $D_k[i][j]$ adalah 0 atau hitam (Zhan *et al.* 2012). Setelah itu ditentukan empat buah titik pada citra yang membentuk sebuah wilayah untuk dilakukan *perspective transformation*. Wilayah yang terbentuk dari empat buah titik ini kemudian diproyeksikan menuju citra berukuran 670 x 600 piksel untuk mendapatkan citra tegak.

Tinggi miniatur bak sampah sesungguhnya adalah 21.2 cm dan tinggi bak sampah pada citra yang telah ditransformasi adalah 600 piksel. Dengan membagi nilai 600 dengan 21.2 dapat diketahui bahwa pada setiap 1 cm terdapat 28.3 piksel. Tinggi timbunan sampah pada citra dapat diketahui dengan melakukan *tracing* dari piksel [0][0] hingga piksel [670][600] untuk mencari nilai piksel 255 pertama. Nilai j pada $[i][j]$ diasumsikan sebagai tinggi timbunan sampah pada citra. Untuk mendapatkan nilai tinggi dalam cm, nilai j dibagi dengan 28.3. Tinggi yang telah didapat dikalikan dengan 30 dan 27 untuk mendapatkan nilai volume timbunan sampah. Untuk memudahkan pembangkitan notifikasi, nilai volume timbunan sampah dibagi menjadi lima level yaitu level 1, 2, 3, 4, dan 5. Volume maksimal bak sampah dibagi lima untuk menentukan *range* antar level. Algoritme pengolahan citra lengkap dapat dilihat pada Gambar 3.

Implementasi

Modul kamera menangkap citra untuk kemudian diolah di dalam Raspberry Pi menggunakan *framework difference algorithm* untuk mendapatkan selisih citra. Citra yang diolah menggunakan *frame difference algorithm* untuk memisahkan *foreground* dari *background*. Algoritme ini membutuhkan dua buah citra sebagai data masukan. Citra pertama digunakan sebagai *background* model yaitu citra ketika bak sampah kosong. Citra kedua adalah citra saat ini yang ditangkap oleh modul kamera Raspberry Pi. Citra dikonversi menuju citra *grayscale* agar *file size* lebih kecil dan mudah diproses untuk tahapan pengolahan citra selanjutnya. *Frame Difference Algorithm* bekerja untuk memisahkan nilai piksel yang mengalami perubahan dari piksel yang memiliki nilai tetap. Piksel-piksel yang mengalami perubahan akan membentuk sebuah citra yang kemudian disebut dengan *foreground*. Citra inilah yang merupakan objek timbunan sampah yang akan dihitung nilai tingginya.



Gambar 3 Proses pengolahan citra.

Noise removal yang digunakan adalah *gaussian blur* sedangkan *otsu binarization* digunakan untuk mengubah citra menjadi citra biner. Metode *otsu binarization* membagi histogram pada citra hasil *grayscale* ke dalam dua daerah yang berbeda tanpa harus memasukkan nilai *threshold* secara manual. Setiap piksel yang memiliki nilai dibawah *threshold* diubah menjadi 0 atau hitam. Sedangkan setiap piksel yang memiliki nilai *threshold* atau lebih diubah menjadi 255 atau putih. Citra yang dihasilkan oleh modul kamera Raspberry Pi masih berbentuk seperti sudut pandang mata burung karena perspektif dari kamera yang menangkap objek dari atas. Agar nilai data tinggi akurat, citra

ditransformasi menggunakan *perspective transformation* untuk mendapatkan pandangan sejajar tegak sehingga lebar, tinggi, dan panjang pada citra sama dengan ukuran pada miniatur bak sampah sesungguhnya. Dibutuhkan empat buah titik untuk membentuk sebuah wilayah yang akan ditransformasi yaitu sisi tegak kardus. Posisi titik-titik ini ditentukan secara manual yaitu di setiap ujung dari sisi kardus yang akan ditransformasi.

Area citra biner yang telah ditransformasi diproyeksikan ke dalam citra 670 x 600 piksel. Diketahui bahwa tinggi bak sampah adalah 21.2 cm dan tinggi citra adalah 600 piksel. Dalam setiap satu cm terdapat 28.3 piksel. Oleh karena itu, tinggi piksel timbulan sampah dibagi dengan angka 28.3 untuk mendapatkan nilai tinggi dalam cm. Potongan kode program dapat dilihat di bawah ini. Volume maksimal bak sampah adalah 17 172 cm kubik. Nilai volume dan level akan dikirimkan menuju server yang telah dikembangkan sebelumnya dan disimpan dalam *database* MongoDB. Jika nilai level sampah adalah 4 atau 5, maka notifikasi akan dibangkitkan oleh aplikasi berbasis *mobile* untuk memberi peringatan agar sampah di TPS segera diangkut. Hasil setiap langkah pengolahan citra dapat dilihat pada Gambar 4.

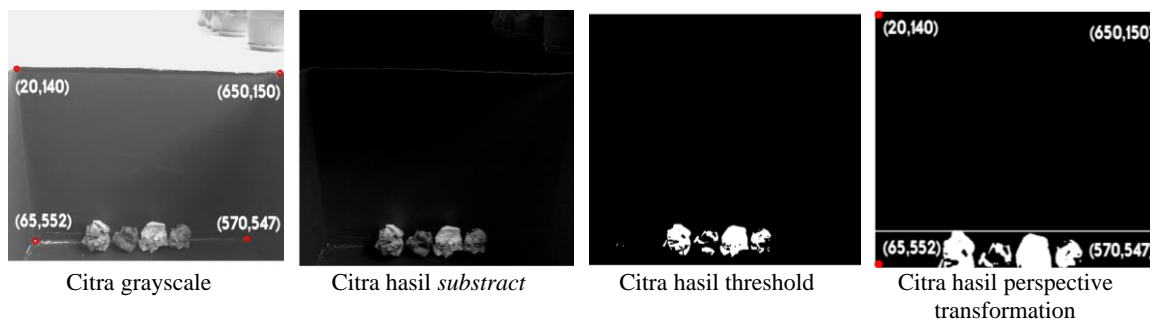
Integrasi

Protokol REST yang digunakan dalam penelitian ini merupakan protokol REST yang telah dikembangkan pada penelitian milik Waladi (2017). API yang dibangun menggunakan bahasa pemrograman javascript, *framework* Keystone.js, dan MongoDB sebagai *database non-relational*. Data volume dan level yang didapatkan dari hasil pengolahan citra oleh Raspberry Pi dikirimkan ke server dengan metode POST. *Id device* dan *id sensornode* yang digunakan merupakan id statis. Data yang diambil dari Raspberry Pi menuju server adalah data vol dan lvl.

Pengujian dan Evaluasi

1 Uji Perhitungan Tinggi dan Pengaruh Intensitas Cahaya

Langkah awal yang dilakukan adalah menguji seberapa akurat modul kamera dan kode program yang digunakan. Enam buah benda dengan ukuran tinggi 3 cm, 6 cm, 9 cm, 12 cm, 15 cm, dan 21 cm dimasukkan ke dalam bak sampah untuk dicari tingginya menggunakan kode program yang telah dibuat. Terdapat dua *dataset*, yaitu nilai aktual (a) atau tinggi hasil pengukuran dengan alat ukur standar dan nilai prediksi (p) atau tinggi hasil pengolahan citra oleh Raspberry Pi. Masing-masing benda diuji sebanyak 10 kali sehingga total data berjumlah 70 buah. *Error* di antara dua *dataset* dapat dihitung dengan menggunakan RMSE. MAPE juga digunakan untuk menghitung nilai akurasi. Data lengkap hasil pengambilan citra dengan 10 kali pengulangan dapat dilihat pada Tabel 2.



Gambar 4 Citra hasil setiap langkah pengolahan citra

Tabel 2 Data perbandingan nilai tinggi diukur dengan penggaris dan Raspberry Pi

No	Aktual (cm)	Prediksi (cm)									
		1	2	3	4	5	6	7	8	9	10
1	3	3.145	3.039	3.216	3.145	2.968	3.216	3.251	3.004	3.145	2.968
2	6	6.290	6.184	6.325	6.254	6.184	6.184	6.290	6.184	6.254	6.148
3	9	9.329	9.258	9.399	9.258	9.223	9.470	9.293	9.329	9.258	9.293
4	12	12.050	12.014	11.908	11.908	12.050	12.120	11.979	12.014	11.873	11.979
5	15	15.124	14.982	15.053	14.947	15.124	14.947	14.947	15.194	15.018	15.159
6	18	18.198	18.375	18.410	18.233	18.445	18.163	18.163	18.375	18.198	18.127
7	21	21.060	21.131	21.166	21.095	21.131	21.201	20.989	21.025	21.095	21.201

Data di atas dimasukkan ke dalam Persamaan 1 dan Persamaan 2 untuk mendapatkan nilai RMSE dan MAPE. Berdasarkan Persamaan 1, n adalah jumlah data yaitu 70 buah. *Error* atau e merupakan hasil pengurangan setiap nilai aktual dengan nilai prediksi. Rumus hitung RMSE dan MAPE dapat dilihat pada Persamaan 3 dan Persamaan 4.

Dari Persamaan 1 didapat nilai RMSE atau *error* antara dua *dataset* yaitu aktual dan prediksi sebesar 0.2060. Angka ini menunjukkan bahwa data memiliki nilai *error* yang cukup besar. Penyebabnya adalah benda uji diletakkan di posisi yang berbeda pada setiap pengulangan. Sedangkan, permukaan dari dasar kardus tidaklah rata, sehingga nilai tinggi memiliki nilai yang berbeda-beda pada setiap pengambilannya. Dari Persamaan 2 diperoleh nilai MAPE sebesar 2.064. Nilai akurasi didapatkan dengan cara mengurangi 100 dengan nilai MAPE atau hasil nilai akurasinya adalah 97.936 persen.

Percobaan juga dilakukan untuk mengetahui pengaruh intensitas cahaya yang ditangkap oleh kamera terhadap keakurasian hasil. Citra diambil di tiga tempat yang berbeda yaitu luar ruangan, dalam ruangan, dan tempat gelap atau minim cahaya. Tinggi timbulan sampah yang digunakan untuk setiap tempat adalah 3 cm, 9 cm, 15 cm, dan 21 cm. Pengambilan citra setiap tinggi timbulan sampah yang berbeda dilakukan sebanyak 10 kali. Total data yang diambil adalah 120 buah citra.

Intensitas cahaya pada ruangan terbuka berubah ubah karena beberapa faktor seperti sinar matahari terhalang awan, cuaca tiba-tiba mendung, dan semakin gelapnya langit seiring waktu. Meskipun intensitas cahaya berubah-ubah, setiap tinggi timbulan sampah yang dihitung tidak mengalami fluktuasi untuk semua pengulangan. Intensitas cahaya di dalam ruangan konstan di setiap pengulangan. Tinggi timbulan sampah yang dihitung juga tidak mengalami perubahan. Percobaan di tempat gelap atau minim cahaya dibagi menjadi dua yaitu timbulan sampah 3 cm dan 9 cm diukur pada intensitas cahaya 0 lx atau gelap dan timbulan sampah 15 cm dan 21 cm diukur pada intensitas cahaya 8 lx atau minim cahaya. Timbulan sampah yang dihitung pada 0 lx menghasilkan nilai 0 cm dan timbulan sampah yang dihitung pada 8 lx menghasilkan nilai tinggi yang fluktuatif atau berubah-ubah. Dapat disimpulkan bahwa semakin kecil intensitas cahaya yang ditangkap oleh modul kamera, semakin sulit citra dideteksi oleh kamera Raspberry Pi. Data hasil percobaan dapat dilihat pada Tabel 3.

2 Uji Fungsional Protokol REST

Server dan REST API yang digunakan pada penelitian ini merupakan adaptasi dari penelitian sebelumnya oleh Waladi (2017). Penelitian ini hanya menggunakan tujuh dari 10 *request* utama dari server yang digunakan. Pengujian setiap fungsi REST perlu dilakukan untuk mengetahui apakah server, *database*, dan aplikasi berbasis *mobile* telah berhasil diintegrasikan dengan sistem yang dibuat pada penelitian ini. Pengujian ini dilakukan dengan menggunakan *software* Postman dengan cara menguji setiap fungsi REST yang dibutuhkan dalam penelitian. Hasil pengujian fungsi REST dapat dilihat pada Tabel 4. Semua fungsi yang digunakan dalam penelitian ini berhasil dijalankan dengan baik.

3 Uji Sistem Pemantauan dan Notifikasi

Aplikasi berbasis *mobile* yang digunakan pada penelitian ini merupakan adaptasi dari penelitian sebelumnya oleh Waladi (2017) dalam menampilkan data hasil pemantauan dari server ke aplikasi berbasis *mobile*. Aplikasi berbasis *mobile* menampilkan data volume dan level yang di GET dari server. Pengujian aplikasi *mobile* untuk visualisasi perlu dilakukan untuk mengetahui apakah aplikasi berbasis *mobile* telah berhasil diintegrasikan dengan sistem yang dibuat pada penelitian ini. Tampilan aplikasi berbasis *mobile* dapat dilihat pada Gambar 5.

Berdasarkan hasil pengujian, aplikasi berbasis *mobile* yang telah dikembangkan dapat berintegrasi dengan baik dengan sistem yang dibuat pada penelitian ini. Pengujian dilakukan dengan menggunakan uji *test case*. Pembangkitan notifikasi terjadi ketika data yang di GET dari server memiliki nilai level 4 atau 5. Notifikasi berupa *floating notification* yang akan menyampaikan pesan bahwa volume timbulan sampah saat ini mencapai *threshold* yaitu empat perlima dari volume maksimal bak sampah. Uji *test case* dapat dilihat pada Tabel 5.

Tabel 3 Pengukuran tinggi menggunakan Raspberry Pi di tiga tempat berbeda

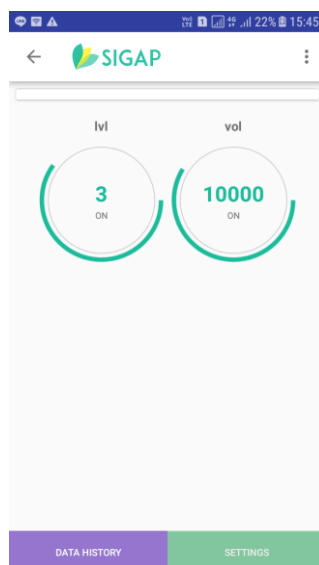
No	Intensitas Cahaya (lx)	Tinggi Timbulan Sampah Diukur dengan Penggaris (cm)	Rata-Rata Tinggi Timbulan Sampah Diukur dengan Raspberry Pi (cm)
1	Luar ruangan		
	475 – 515	3	3.2155 ± 0.0000
	308 – 331	9	9.0458 ± 0.0003
	219 – 242	15	14.9311 ± 0.1620
2	Dalam ruangan		
	151 – 179	21	21.0919 ± 0.0112
	39	3	3,1095 ± 0.0000
	30	9	8,9753 ± 0.0000
3	Gelap atau remang	32	15,053 ± 0.0000
		35	20,9541 ± 0.0000
		0	0 ± 0.0000
		0	0 ± 0.0000
		8	14,6290 ± 0.3130
		8	20,6363 ± 0.0625

Tabel 4 Pengujian fungsi REST

Fungsi REST	Hasil yang diharapkan	Hasil yang didapatkan	Status
GET Device	Server akan mengembalikan daftar <i>device</i>	Server mengembalikan daftar <i>device</i>	Berhasil
POST Device	Server akan menyimpan <i>device</i> baru ke <i>database</i>	Server menyimpan <i>device</i> baru ke <i>database</i>	Berhasil
GET Sensornode	Server akan mengembalikan daftar <i>sensornode</i>	Server mengembalikan daftar <i>sensornode</i>	Berhasil
POST Sensornode	Server akan menyimpan <i>sensornode</i> baru ke dalam <i>database</i>	Server menyimpan <i>sensornode</i> baru ke dalam <i>database</i>	Berhasil
GET Dataset	Server akan mengembalikan data sensor	mengembalikan data sensor	Berhasil
POST Dataset	Server akan menyimpan data baru ke <i>database</i>	Server menyimpan data baru ke <i>database</i>	Berhasil
POST Notifikasi	Server akan membangkitkan notifikasi	Server membangkitkan notifikasi	Berhasil

SIMPULAN

Penelitian ini telah mampu mengimplementasikan *framework difference algorithm* dan *perspective transformation* untuk mencari volume timbulan sampah dengan menggunakan modul kamera Raspberry Pi. Perhitungan pada Raspberry Pi memiliki nilai RMSE sebesar 0.2060 dan akurasi sebesar 97.93 persen. Salah satu faktor keakuratan ditentukan oleh banyak sedikitnya intensitas cahaya yang ditangkap oleh kamera. Semakin sedikit cahaya yang diterima oleh kamera, semakin besar kemungkinan fluktuasi terjadi. Faktor lain yang mempengaruhi hasil perhitungan tinggi dan volume timbulan sampah adalah posisi kamera dan warna sampah.

Gambar 5 Tampilan pemantauan aplikasi berbasis *mobile*

Tabel 5 Uji test case

Fungsi	Status
Visualisasi data volume saat nilai level 1	Berhasil
Visualisasi data volume saat nilai level 2	Berhasil
Visualisasi data volume saat nilai level 3	Berhasil
Visualisasi data volume saat nilai level 4	Berhasil
Visualisasi data volume saat nilai level 5	Berhasil
Tidak ada notifikasi saat nilai level 1	Berhasil
Tidak ada notifikasi saat nilai level 2	Berhasil
Tidak ada notifikasi saat nilai level 3	Berhasil
Notifikasi saat nilai level 4	Berhasil
Notifikasi saat nilai level 5	Berhasil

Penelitian ini juga telah mampu menerapkan sebuah protokol pengiriman dan pengambilan data berbasis REST dengan tujuh *request* utama yang diadaptasi dari penelitian sebelumnya yaitu protokol REST pada pengendali otomatis irigasi oleh Waladi (2017) dengan menggunakan MongoDB sebagai *database* yang digunakan. Selain itu, visualisasi data dan pembangkitan notifikasi juga telah diterapkan dengan menggunakan aplikasi berbasis *mobile*.

Sistem dapat dikembangkan lagi menjadi sistem pemantauan volume timbulan sampah yang menggunakan beberapa *node* Raspberry Pi. Selain itu, teknik *perspective transformation* yang digunakan saat ini masih dilakukan secara manual dengan mencari satu persatu titik wilayah yang akan ditransformasi. Penelitiannya selanjutnya diharapkan dapat mengimplementasikan transformasi perspektif secara otomatis sehingga dapat mempercepat proses pengolahan data.

DAFTAR PUSTAKA

- Almuqaddami Y. 2017. Implementasi filter kalman pada sistem tertanam waktu nyata untuk menghitung jumlah kendaraan di jalan raya [Skripsi]. Bogor (ID): Institut Pertanian Bogor.
- [Bappeda] Badan Perencanaan Pembangunan Daerah. 2014. *Laporan Akhir Penyusunan Layanan Persampahan Kota Bogor*. Bogor (ID): Bappeda Kota Bogor.
- [BPS] Badan Pusat Statistik. 2018. Jumlah Penduduk dan Laju Pertumbuhan Penduduk Menurut Kecamatan di Kota Bogor 2010, 2016, dan 2017 [Internet]. [diakses pada 2018 Nov 1]; Tersedia pada: <https://bogorkota.bps.go.id>

- Cohen B. 2012. What exactly is a smart city? [Internet]. [diakses pada 2018 Nov 1]; Tersedia pada: <https://www.fastcompany.com/user/boyd-cohen>
- Damanhuri E. 2006. Teknologi dan pengelolaan sampah kota di Indonesia. Di dalam: *Workshop Nasional Biokonversi Limbah*; Malang, 2006 Apr 11. Malang(ID): Universitas Brawijaya.
- [Diskominfostandi] Dinas Komunikasi, Informatika, Statistik dan Persandian. 2017. *Buku #2 Masterplan Smart City Kota Bogor 2017-2021*. Bogor (ID): Diskominfostandi Kota Bogor.
- [DLH] Dinas Lingkungan Hidup. 2015. *Laporan Pengelolaan Lingkungan Hidup Kota Bogor Tahun 2015*. Bogor (ID): DLH Kota Bogor.
- [DLH] Dinas Lingkungan Hidup. 2017. *Dokumentasi Informasi Kinerja Pengelolaan Lingkungan Hidup Daerah Kota Bogor 2017*. Bogor (ID): DLH Kota Bogor.
- El-Haggar SM. 2007. *Sustainable Industrial Design and Waste Management: Cradle-to-cradle for Sustainable Development*. London (UK): Elsevier Academic Press.
- Hariyanto A, Wibowo A, Noertjahyana A. 2013. Sistem manajemen skripsi program manajemen bisnis berbasis web service dan phonegap. *Jurnal Infra*. 1 (2): 329-334.
- Rusmaya D, Rochaeni A, Mulyana H. 2019. Perencanaan Jalur Pengangkutan Sampah di Kabupaten Bogor, Kota Bogor, dan Kota Depok menuju Stasiun Pengumpul Antara (SPA). *Journal of Community Based Environmental Engineering and Management*. 2(1): pp.1-8.
- Samsuri, Maulana LH. 2019. Model Pengelolaan Sampah Perkotaan (Survey Pada Pengelolaan Persampahan Kota Bogor). *Jurnal Visionida*. 5(2): pp.54-61.
- Singla N. 2014. Motion detection based on frame difference method. *IJICT*. 4 (15): 1559-1565.
- Uckelmann D, Harrison M, Michahelles F. 2011. *Architecting the Internet of Things*. London (UK): Springer Heidelberg Dordrecht.
- Waladi A. 2017. Penerapan konsep *internet of things* pada sistem pengendali otomatis irigasi lahan produksi [Skripsi]. Bogor (ID): Institut Pertanian Bogor.
- Zhan C, Duan X, Xu S, Song Z, Luo M. 2007. An improved moving object detection algorithm based on frame difference and edge detection. Di dalam: *Fourth International Conference on Image and Graphics, ICIG 2007*; Chengdu, 2007 Agu 22-24. Chengdu(CN): IEEE. doi: 10.1109/ICIG.2007.153