

Analisis Performansi Ultrasound Vehicle Counter menggunakan Algoritma Normalized Auto-Correlation

Deta Kurnia Soundra¹, Aji Gautama Putrada², Maman Abdurrohman³

^{1,2,3}Fakultas Informatika, Universitas Telkom, Bandung

¹soundra@students.telkomuniversity.ac.id, ²ajigps@telkomuniversity.ac.id,

³abdurohman@telkomuniversity.ac.id,

Abstrak

System lalu lintas pada saat ini telah menunjukkan evolusi seiring berkembangnya teknologi. Di dukung dengan system transportasi yang tentunya sudah canggih. Dua system tersebut merupakan bagian dari smart city yang diterapkan di kota-kota besar. Pada dasarnya semua saling berkomunikasi untuk terciptanya smart city yang terintegrasi. Akan tetapi komunikasi tersebut harus lah sesuai dengan waktu nyata sehingga semua komponen smart city saling terkoneksi secara real-time. Dalam hal ini peneliti merancang sebuah system penghitung kendaraan secara real-time yang dapat menghitung kendaraan yang melintas pada suatu ruas jalan. Penerapan dan pengembangan sensor Ultrasound, mikrokontroler, mikroprosesor dan Internet Of Things yang saling terhubung untuk memantau kondisi jalan. Dengan menggunakan algoritma Normalized Auto-Correlation yang efisien dalam komputasi untuk mendeteksi kendaraan yang melintas. Algoritma Normalized Auto-Correlation biasanya digunakan untuk membandingkan 2 buah sinyal frekuensi untuk menentukan kemiripan dalam sinyal tersebut. Dalam hal ini algoritma Normalized Auto-Correlation digunakan untuk menentukan perhitungan kendaran yang melewati sensor ultrasound. Sistem akan mendeteksi dengan membandingkan data input dari sensor ultrasound dengan cara membuat data sample terlebih dahulu kemudian data sample tersebut dibandingkan dengan data setelah data sample tersebut. Setelah itu akan keluar nilai korelasinya yang sudah di normalisasikan dengan skala 0-1.0. Dari hasil korelasi tersebut peneliti menentukan threshold untuk perhitungan kendaraan, pada saat nilai korelasi tersebut < 0.70 maka system akan menyatakan dan menghitung sebuah kendaraan telah melewati sensor. Hasil dari analisis pengujian diperoleh persentase sebesar 10.09%.

Kata kunci : Real-time sensing, Smart City, Smart Transportation, Smart Traffic System, Auto-Correlation, Ultrasound

Abstract

The current traffic system has addressed the transition to the completion of technology. Supported by a complete transportation system. This two system is part of a smart city that is applied in big cities. Basically all communicate with each other to create an integrated smart city. However, this communication must be in accordance with the real time so that all smart city components are connected in real-time. In this case the researcher approves a real-time vehicle counting system that can calculate vehicles passing on a road segment. Application and development of Ultrasound sensors, microcontrollers, microprocessors and Internet Of Things that are interconnected to improve road conditions. By using the Normalized Auto-Correlation algorithm that is efficient in computing for the protection of passing vehicles. The Auto-Correlation Normalization Algorithm is usually used to compare two frequency signals to determine the similarity in the signal. In this case the Automatic Correlation-Normalization algorithm is used to determine vehicle calculations that pass the ultrasound sensor. The system will compare by comparing input data from ultrasound sensors by making sample data first then the sample data is compared with the data after the sample data. After that the correlation value will come out which has been normalized on a scale of 0-1.0. From the results of the assessment the researcher determines the threshold for the calculation of the vehicle, when the estimated value is <0.70, the system will approve and calculate the vehicle that has passed the sensor. The result from analysis of the test that percentage of error reaching 10.09% from the test.

Keywords: Real-time sensing, Smart City, Smart Transport, Smart Traffic System, Auto-Correlation, Ultrasound

1. Pendahuluan

Latar Belakang

Kemacetan yang terjadi khususnya di kota-kota besar saat ini merupakan salah satu masalah yang sangat umum terjadi dan dijumpai sepanjang sudut kota. Kemacetan ini terjadi karena terus bertambahnya jumlah kendaraan setiap tahunnya. Disamping bertambahnya kepadatan kendaraan, perkembangan infrastruktur jalan di kota-kota besar tidak lah berkembang secepat pertumbuhan populasi kendaraan yang saat ini semakin banyak dan semakin padat. Karena keterbatasan lahan adalah alasan utama kurang berkembangnya infrastruktur, sehingga kemacetan seakan terus menerus tidak tertangani. Selain memperbaharui jaringan jalan, pengembangan *smart transportation systems* harus lah menjadi prioritas dalam mengatasi kemacetan dan kepadatan lalu lintas. Salah satu komponen dari *smart transportation systems* adalah sistem penghitungan kendaraan, yang digunakan untuk mengumpulkan informasi arus lalu lintas. Pengembangan sistem perhitungan kendaraan ini harus memiliki beberapa persyaratan, seperti komponen alat tersebut harus kuat dalam berbagai kondisi cuaca dan cahaya, harus dapat bekerja dalam waktu 24 jam *non-stop* dan mengumpulkan data secara real-time serta konsumsi daya sistem yang harus rendah.

Saat ini telah banyak sistem yang bisa menghitung jumlah kendaraan pada sepanjang jalan. [1] Di beberapa kota telah menerapkan sistem penghitungan kendaraan dengan menggunakan loop induktif dengan akurasi yang sangat tinggi, namun dalam hal pemeliharaan membutuhkan biaya yang tinggi. Perangkat sensor Ultrasonik bisa digunakan sebagai *alternative* dalam membangun sistem penghitungan kendaraan. Sensor Ultrasonik memiliki kapabilitas dalam situasi minim cahaya, sehingga bisa digunakan pada malam hari dan cost yang tidak terlalu besar. Dalam pengembangan sistem penghitungan kendaraan sensor ultrasound dikombinasikan dengan sebuah mikrokontroler yang terhubung dengan jaringan internet. Sistem tersebut berbasis IoT yang dihubungkan melalui protocol MQTT dengan menggunakan *broker* Raspberry Pi. Sistem *local* tersebut saling terintegrasi, dengan menggunakan algoritma *Normalized Auto-Correlation* untuk melakukan klasifikasi pada saat sensor ultrasound mengirimkan data. Algoritma tersebut bisa diterapkan dalam sistem ini karena data dari sensor ultrasound yang berupa sinyal frekuensi gema yang bisa menentukan jarak. Jarak tersebut dikumpulkan menjadi suatu data sample, kemudian data sample pertama dibandingkan dengan data selanjutnya setelah sample untuk mencari kemiripan.

Topik dan Batasannya

Berdasarkan latar belakang, dapat diidentifikasi masalah yang diantaranya bagaimana sistem penghitungan kendaraan menggunakan sensor ultrasound dapat menghitung kendaraan yang melewati jalur tertentu serta bagaimana performansi metode klasifikasi algoritma *Normalized Auto-correlation* dalam mendeteksi kendaraan yang melintas.

Batasan masalah yang terdapat pada penelitian ini adalah ketika dua kendaraan melaju bersamaan maka akan ada yang tertutupi dan tidak terdeteksi, penelitian hanya dilakukan dipersimpangan gerbang tol Buah Batu, dalam hal penghitungan kendaraan hanya roda empat ke-atas yang menjadi objek penelitian.

Tujuan

Tujuan dari identifikasi masalah pada penelitian ini adalah merancang sebuah system yang dapat menghitung kendaraan yang melewati jalur menggunakan sensor ultrasound serta menerapkan algoritma *Normalized Auto-Correlation* dan menganalisis hasil dari pengujian alat tersebut. Tujuan dari penelitian ini dapat dilihat pada Tabel 1 yaitu :

Tabel 1. Keterkaitan antara tujuan, pengujian dan kesimpulan

No	Tujuan	Pengujian	Kesimpulan
1	Merancang dan membuat perangkat system penghitungan kendaraan	Menjalankan perangkat untuk mendeteksi dan menghitung jumlah kendaraan yang melewati jalur dan melakukan pengiriman data menggunakan protokol MQTT	Perangkat berfungsi dengan baik dan data dapat di kirim menggunakan Protokol MQTT
2	Menerapkan algoritma <i>Normalized Auto-Correlation</i> dalam menentukan perhitungan kendaraan	Menguji algoritma <i>Normalized Auto-Correlation</i> dengan data input dari sensor ultrasound untuk menentukan kendaraan yg melintas dan menghitung jumlah kendaraan yang melintas.	Algoritma <i>Normalized Auto-Correlation</i> dapat menentukan kendaraan yang melintas dan menghitungnya.
3	Memberikan informasi secara real-time kepada user	Menguji apakah data tersebut dapat di monitor secara langsung pada Node-Red Dashboard	User dapat memonitor jumlah kendaraan yang melintas

Organisasi Tulisan

Penulisan bab pertama membahas mengenai masalah dan batasan serta tujuan dari penelitian ini. Pada bab dua dibahas studi literatur yang digunakan sebagai bahan informasi dan referensi untuk perancangan sistem penghitungan kendaraan. Pada bab tiga dijelaskan rancangan sistem secara umum dan analisis kebutuhan pada sistem. Penulisan pada bab empat dijelaskan hasil pengujian dan analisis sistem. Untuk penulisan bab lima dijelaskan kesimpulan selama penelitian berlangsung dan saran yang diberikan.

2. Studi Terkait

Dalam penelitian [2] yang berjudul “*Dynamic Traffic Management System using Infrared and Internet of Things*” pada tahun 2017, telah berhasil menggabungkan teknologi dengan menggunakan Raspberry Pi, sensor PIR dan *Image Processing*. Raspberry PI berperan sebagai peralatan kontrol yang dapat menerima inputan dari sensor PIR yang terdapat pada jalur persimpangan. Pada penelitian tersebut, lampu lalu lintas diubah menjadi lampu lalu lintas pintar yang bisa merubah waktu dinamis dengan cara mendeteksi jalur dengan image processing. Sensor tersebut bisa mendeteksi kepadatan jalur dan merubah waktu lampu merah dan hijau ketika kepadatan berubah. Sistem akan mendeteksi kendaraan menggunakan sensor ultrasound yang digunakan dalam penghitungan kendaraan disetiap jalur. Setiap kali sensor ultrasound mendeteksi bagian kendaraan di jalan, sistem mengukur jarak kendaraan yang sesuai berdasarkan lokasi jalurnya.

[3] Deteksi kendaraan dan klasifikasi jalur dapat dilakukan dengan menggunakan data pada jarak ke kendaraan dan waktu yang dibutuhkan kendaraan untuk melewati rentang deteksi sensor ultrasound. Algoritma pendeteksian terdiri dari tiga bagian. Bagian pertama menghitung ambang, yang merupakan titik dimana kendaraan terdeteksi di setiap jalur. Bagian kedua menyaring data yang tidak perlu, seperti kebisingan dari lingkungan alami. Bagian ketiga menentukan lokasi kendaraan di banyak lajur dan menghitung volume lalu lintas, berdasarkan data yang disaring dan ambang yang dihitung.

2.1 Auto-Correlation

[4] Auto-Correlation adalah perbandingan deret waktu dengan dirinya sendiri pada waktu yang berbeda. Ini bertujuan, misalnya, untuk mendeteksi pola berulang atau musiman. Biasanya digunakan untuk mencari kemiripan frekuensi pada sinyal radio dan sinyal telekomunikasi lainnya untuk menentukan pola apakah yang ada dalam sinyal tersebut. Namun pada penelitian ini menggunakan *Normalized Auto-Correlation* yang mana bisa membandingkan dua jenis sinyal dari waktu yang berbeda dengan menggunakan scoring berbeda. Metode ini dapat membandingkan sinyal dengan rentang nilai yang berbeda. Gagasan di balik konsep autokorelasi adalah untuk menghitung koefisien korelasi dari deret waktu dengan dirinya sendiri, bergeser waktu. Jika data memiliki periodisitas, koefisien korelasi akan lebih tinggi ketika kedua periode beresonansi satu sama lain.

Langkah pertama adalah mendefinisikan operator untuk menggeser deret waktu dalam waktu, menyebabkan penundaan t . Ini dikenal sebagai operator lag:

$$\text{lag}(X_i, t) = X_{i-t}$$

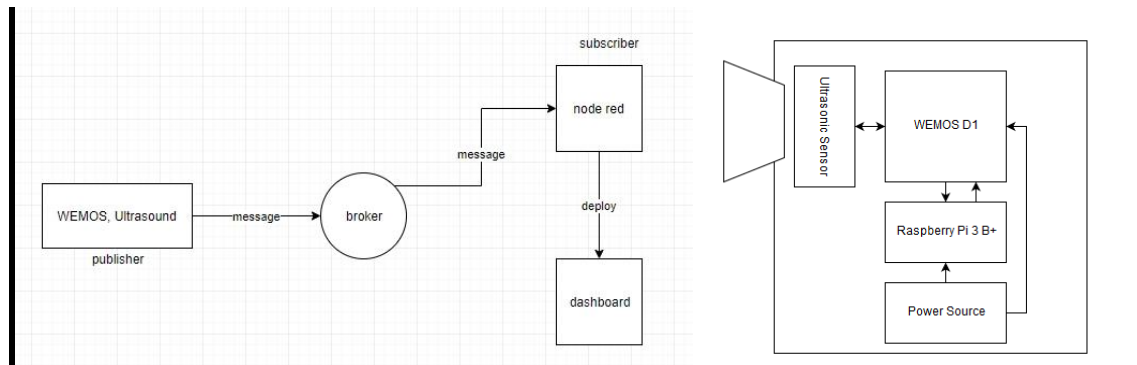
autokorelasi pada time series dengan lag t didefinisikan sebagai:

$$\text{auto_corr}(X, t) = \text{corr}[X, \text{lag}(x, t)]$$

Berikut adalah formula untuk normalisasinya :

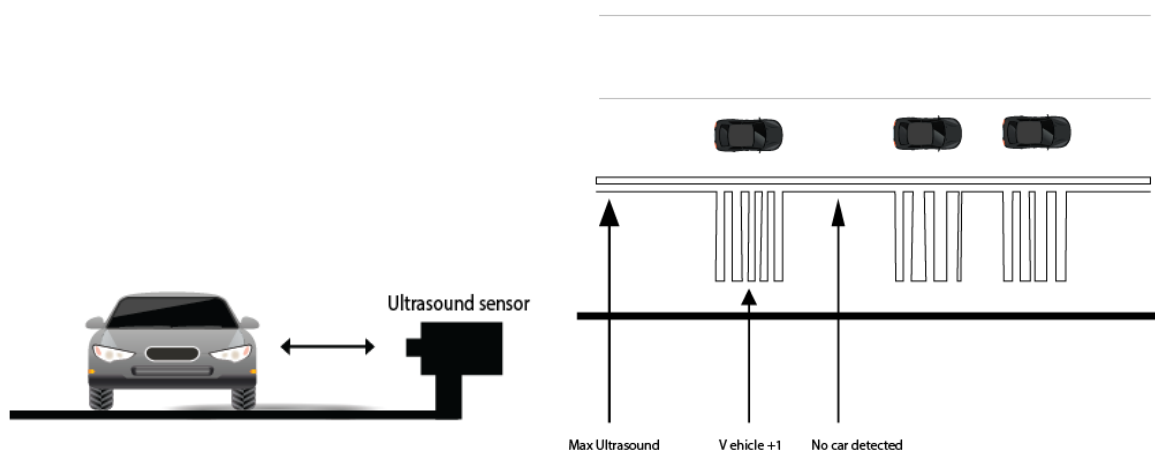
$$\text{norm_auto}(x, y) = \frac{\sum_{n=0}^{n-1} x[n] * y[n]}{\sqrt{\sum_{n=0}^{n-1} x[n]^2 * \sum_{n=0}^{n-1} y[n]^2}}$$

3. Sistem Perhitungan Kendaraan



Gambar 1. Gambaran Umum Sistem dan gambaran alat

Pada gambar 1 dijelaskan system perhitungan kendaraan memiliki output berupa hasil perhitungan kendaraan yang melewati sensor. Pada system ini terdapat satu sensor ultrasound yang memiliki fungsi untuk mendeteksi jarak kendaraan. Mikrokontroler berfungsi untuk memproses data yang ada dan kemudian mengunggah data tersebut melalui protokol MQTT. Data sekaligus diproses bersama pada board mikrokontroler. Kemudian data hasil perhitungan kendaraan akan ditampilkan pada node-red dashboard.



Gambar 2. Konfigurasi Posisi dan data hasil sensor ultrasound

Pada gambar 2 menunjukkan posisi alat penghitung kendaraan yang disimpan pada trotoar untuk mendeteksi adanya kendaraan lewat. Kemudian sensor ultrasound tersebut akan menghitung jumlah kendaraan yang melintas di depannya. Pada gambar 2 juga di perlihatkan bagaimana sensor mendeteksi kendaraan yang melintas, kemudian menghitungnya pada saat mobil terdeteksi dan melintas.

3.1 Kebutuhan Sistem

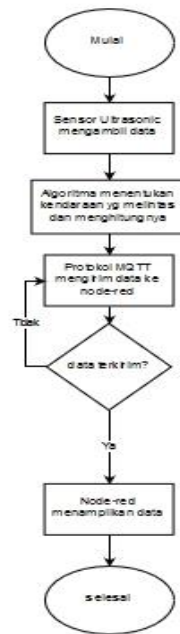
Sistem penghitungan kendaraan memiliki fungsionalitas sebagai berikut:

- Sistem mampu mendeteksi kendaraan yang melintas dan menghitungnya.
- Sistem dapat terintegrasi dengan protokol MQTT untuk mengolah nilai input dari sensor dengan metode *Normalized Auto-Correlation*.

- Sistem dapat menampilkan jumlah kendaraan yang melintas kemudian ditampilkan pada sebuah UI untuk dapat dimonitor secara langsung.

3.2 Alur Diagram pada Sistem

Terdapat alur diagram system yang menjelaskan bagaimana cara kerja system secara bertahap. Alur diagram system dapat dilihat Pada Gambar 2 :



Gambar 3. Alur Diagram Sistem

Pada gambar 3 di atas adalah alur dari algoritma sistem pada perangkat penghitung kendaraan. Pertama sensor membaca dan mengambil data dengan cara mengukur jarak antara sensor dan benda yang melintas. Kemudian algoritma *Normalized Auto-Correlation* akan memproses data hasil inputan dari sensor kemudian data akan di normalisasi. Selanjutnya pengiriman data melalui protokol MQTT yang akan ditampilkan pada *dashboard* node-red.

3.3 Spesifikasi Sistem

Pada penelitian yang dilakukan, peneliti menggunakan perangkat keras dan perangkat lunak dengan fungsionalitas yang akan dijelaskan pada tabel 2 sebagai berikut :

Tabel 2. Spesifikasi Sistem

Jenis Perangkat	Fungsionalitas
Wemos D1	Menerima data dari sensor
Wemos D1	Mengirim data yang diterima dari arduino ke internet.
Arduino IDE	Perangkat Lunak yang digunakan untuk melakukan pengaturan untuk arduino dan wemos D1
Raspberry Pi 3 B+	Sebagai broker MQTT untuk mengirimkan data dan menyimpan data
Sensor Ultrasonik hcsr 04	Pendeteksi jarak kendaraan
MQTT Protocol	Protokol pengiriman data yang digunakan untuk mem- <i>publish</i> dan men- <i>subscribe</i>
Node-Red	<i>Platform</i> yang digunakan untuk <i>monitoring</i> dan menyimpan data

4. Evaluasi

4.1 Skenario Pengujian

Pada penelitian ini akan menguji fungsionalitas dari system dan menganalisis performa dari algoritma *auto-correlation* yang ada pada sistem tersebut. Berikut merupakan pengujian yang dilakukan dalam penelitian :

1. Pengujian rangkaian system

Penelitian ini menggunakan 1 buah sensor ultrasound hsrc 04 dihubungkan dengan Wemos D1, setelah semua perangkat terpasang lalu dilakukan pengaturan sehingga data yang ada dapat terunggah melalui MQTT.

2. Pengujian Algoritma Normalized Auto-Correlation

Penelitian ini akan menguji akurasi dari algoritma *Auto-correlation* dalam menentukan dan menghitung jumlah kendaraan yang melintas pada jalur tersebut.

3. Pengujian pengiriman data melalui *protocol* MQTT dan NodeRed

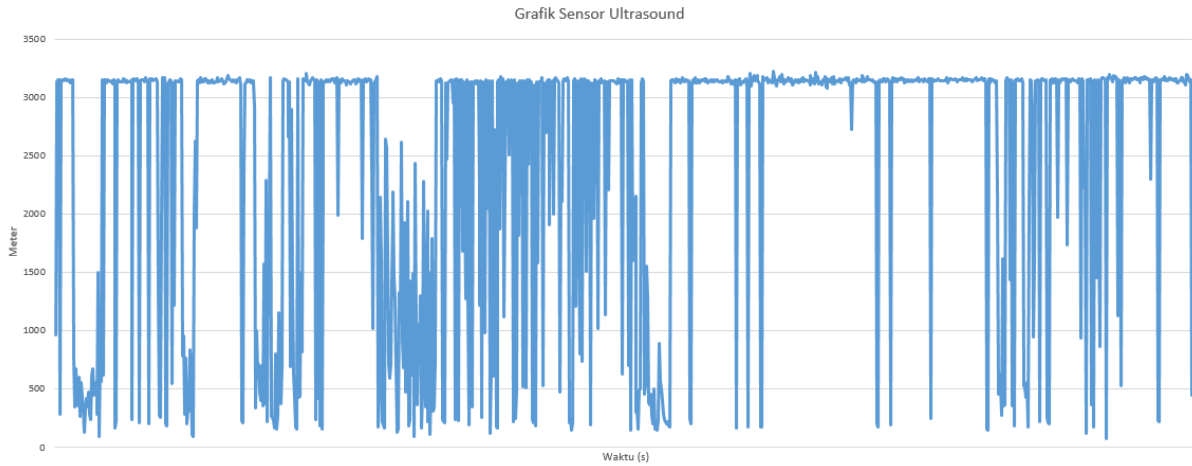
Penelitian ini akan menguji system dalam mengirimkan data kemudan ditampilkan pada laman dashboard Nodered.

4. Menganalisis performansi algoritma *auto-correlation*.

Untuk mengetahui performansi dari algoritma *auto-correlation* maka dilakukan penghitungan jumlah *error count* pada data yang telah diterima dengan cara membandingkan data yang aktual dan data yang diterima oleh sensor.

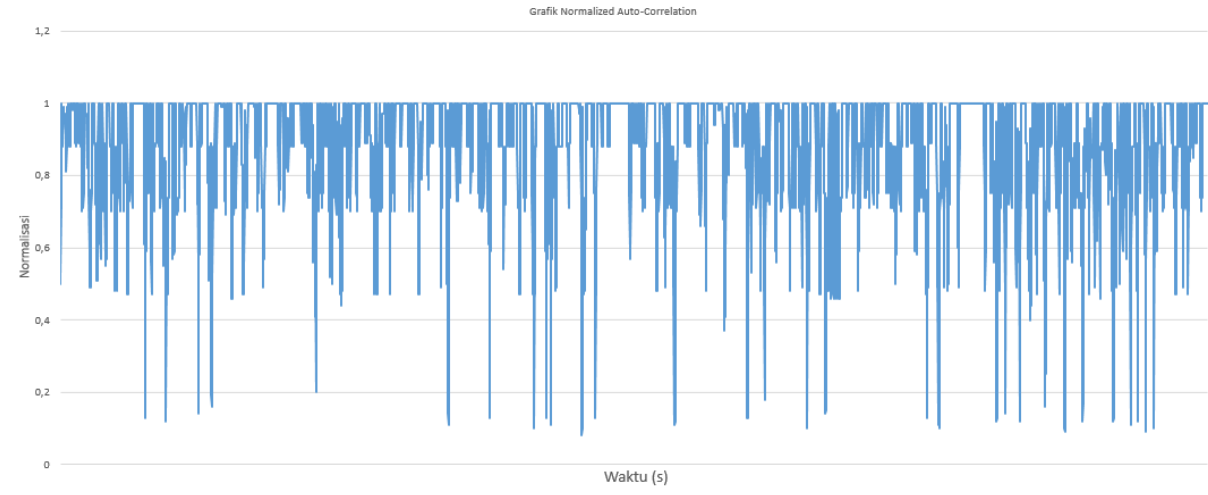
4.2 Hasil Pengujian dan Analisis

Dari pengujian yang telah dilakukan didapatkan data grafik hasil sensing oleh sensor ultrasound yang berupa data jarak pantulan sonar. Data tersebut belum diproses oleh algoritma *Normalized Auto-Correlation* sehingga hanya data jarak yang dihasilkan. Berikut adalah grafik hasil sensing oleh sensor ultrasound :



Grafik 1. Data hasil sensing

Pada penelitian ini sensor ultrasound di ujicoba terlebih dahulu tanpa menggunakan algoritma *Normalized auto-Correlation*. Setelah data hasil inputan sensor telah di uji coba, maka penelitian dilanjutkan untuk menguji algoritma *Normalized Auto-correlation* pada sistem. Dan berikut adalah hasil pengambilan data dalam 1 jam :



Grafik 2. Data hasil proses algoritma normalisasi

Dari grafik di atas bisa di lihat perbandingan menggunakan algoritma *normalized auto-Correlation* dengan tanpa menggunakan algoritma *normalized auto-Correlation*. Hasil dari grafik tersebut menunjukkan bahwa dengan menggunakan algoritma *normalized auto-Correlation* data hasil inputan tersebut sudah di organisir dengan menentukan skala skor korelasinya antara 0-1. Jika tidak menggunakan algoritma *normalized auto-Correlation* maka data hasil inputan akan banyak bervariasi sehingga tidak bisa menentukan kapan harus menghitung kendaraan.

Data tersebut diambil di jalan gerbang tol Buah Batu, pada saat ada kendaraan melintas dengan rata-rata kecepatan kendaraan berkisar antara 40-60 Km/Jam. Kemudian setelah kendaraan melintas jika skor normalisasinya < 70 maka akan terhitung dalam counter + 1. Setelah uji coba selesai maka didapatkan hasil penghitungan jumlah kendaraan yang melintas pada tabel analisis dibawah :

Waktu	Actual passing	Detected passing	Error rate	Missing	Overcount
06.00-06.10	113	121	7.0	3	11
06.10-06.20	142	134	5.6	6	14
06.20-06.30	106	97	8.4	2	11
06.30-06.40	89	103	3.5	0	14
06.40-06.50	75	123	6.4	0	48
total	525	578	61,8	11	98

Tabel 3. Data hasil penghitungan kendaraan yang melintas

Dari hasil tabel pengujian diatas didapatkan bahwa dari $(578-525)/525 \times 100\%$ maka Persentase *Error* yang di dapatkan adalah 10.09 %. Hasil tersebut di peroleh dari data *actual* yang di hitung secara manual kemudian dibandingkan dengan data hasil dari sensor ultrasound tersebut. Kemudian perhitungan persentase *error* di atas merupakan pembacaan sensor yang tidak benar dengan perhitungan actual, diantaranya adalah *missing count* dan *overcount*. Maka diperoleh persentase *error* dari beberapa waktu, kemudian rata rata dari persentase *error* tersebut di jumlahkan dan mendapatkan nilai persentasenya.

5. Kesimpulan

Kesimpulan yang didapat dari penelitian ini adalah penggunaan *auto-correlation* berfungsi cukup baik untuk system penghitungan kendaraan dengan mempunyai Persentase *Error* yang mencapai 10.09 % . Di katakan cukup baik karena berdasarkan penelitian sebelumnya yang memperoleh persentase *error* pada 3% maka sistem ini bisa dikatakan mendekati sehingga bisa di katakan cukup baik. Sistem juga bisa berfungsi dan terhubung dengan platform Node-red melalui protocol MQTT, dan perhitungan dilakukan secara realtime.

Saran yang dapat diberikan setelah melakukan penelitian ini adalah penelitian ini dapat di implementasikan kepada objek penelitan yang cakupannya lebih besar seperti *Smart Traffic System* dan objek *Smart City* lainnya. Alat ini mempunyai peran dalam pembangunan sistem terintegrasi pada Smart Traffic Light, yaitu alat ini bisa mengukur kepadatan kendaraan pada suatu kota secara *realtime*. Selain itu juga dapat menggunakan metode algoritma *machine learning* yang lain sebagai perbandingan dan mencari tingkat akurasi yang lebih besar.

Daftar Pustaka

- [1] Cheung, S.Y.; Varaiya, P.P. Traffic Surveillance by Wireless Sensor Networks: Final Report; California PATH Research Report UCB-ITS-PRR-2007-4; University of California: Berkeley, CA, USA, 2007. Available online: <http://www.its.berkeley.edu/publications/UCB/2007/PRR/UCBITS-PRR-2007-4.pdf>
- [2] Paul Jasmine Rani, Khoushik Kumar, Naresh, Vignesh. 2017. Dynamic Traffic Management System using infrared and Internet of Things. Department of Computer Science Rajalakshmi Insitute of Technology.
- [3] Soobin Jeon, Eil Kwon and Inbum Jung. 2014 . Traffic Measurement on Multiple Drive Lanes with Wireless Ultrasound Sensors Department of Computer Information and Communication Engineering, Kangwon National University, Chuncheon
- [4] The Autocorrelation Function. 2016. Algoritma Auto-Correlation. Di ambil dari <https://www.alanzucconi.com/2016/06/06/autocorrelation-function/> (diakses 12 January 2019),