

Pengenalan Aktivitas Berkendara Berbasis Sensor *Accelerometer* Pada *Smartphone* Dengan Fitur *Auto Feedback* Menggunakan Algoritma *k-Nearest Neighbour*

Tri Ramadani Arjo¹, Reipki Anbeni²

Teknik Multimedia Jaringan, Politeknik Negeri Batam
ramadani@polibatam.ac.id¹, reipki.anbeni@gmail.com²

Article Info

Article history:

Received 2018-04-30

Revised 2018-06-12

Accepted 2018-07-01

Keyword:

Accelerometer, kecelakaan lalu lintas, *K-Nearest Neighbour*, *KNN*, *smartphone*.

ABSTRAK

Penggunaan handphone merupakan salah satu penyebab kecelakaan lalu lintas. Hal ini sangat memprihatinkan mengingat penggunaan handphone merupakan salah satu aktivitas manusia yang saat ini sering dilakukan. Penelitian ini berusaha memberikan solusi untuk mengurangi penyebab kecelakaan lalu lintas yang disebabkan penggunaan handphone saat sedang berkendara dengan cara memanfaatkan sensor *accelerometer* yang terdapat pada *smartphone*. Data pada sensor *accelerometer* diklasifikasikan dengan menggunakan algoritma *K-Nearest Neighbour* untuk mengetahui apakah pengguna handphone sedang berkendara atau sedang tidak berkendara. Penelitian ini juga membuat aplikasi *auto feedback* yang dapat membalas pesan secara otomatis apabila pengguna *smartphone* sedang berkendara sehingga diharapkan dapat menciptakan kondisi berkendara yang aman. Dalam rangka mengetahui ketepatan algoritma dalam mengklasifikasikan aktivitas berkendara, maka dilakukan pengujian pada aplikasi *auto feedback* dengan cara mengirimkan beberapa pesan saat pengguna handphone sedang berkendara dan saat sedang tidak berkendara.

Copyright © 2017 Journal of Applied Informatics and Computing.
All rights reserved.

I. PENDAHULUAN

Berdasarkan data pada Global Status Report on Road Safety 2015 yang dirilis oleh WHO (World Health Organization), Indonesia menempati urutan ke-4 dunia dengan angka kecelakaan lalu lintas berkendara [1]. Salah satu penyebab kecelakaan dalam berkendara disebabkan oleh penggunaan telepon pintar secara berlebihan seperti adanya telepon masuk, pesan masuk, bahkan notifikasi sosial media secara massif dengan tingkat kepentingan yang berbeda. Penggunaan telepon pintar ketika berkendara menjadi masalah yang serius dan harus dikurangi, mengingat saat ini penggunaan telepon pintar merupakan salah satu aktivitas yang sering dilakukan oleh manusia.

Di tengah perkembangan teknologi yang semakin maju, aktivitas manusia tidak hanya dapat dikenali oleh manusia itu sendiri. Kini aktivitas manusia juga dapat dikenali oleh sensor. Salah satu sensor yang dapat digunakan untuk mengenali aktivitas manusia yang ditanamkan pada telepon pintar adalah sensor *accelerometer*. Data pada sensor *accelerometer* akan menunjukkan pola gerakan yang

disebabkan oleh aktivitas manusia dalam jangka waktu tertentu.

Agar dapat mengklasifikasikan sebuah gerakan yang dihasilkan dari sensor *accelerometer* ke dalam aktivitas manusia, diperlukan sebuah algoritma klasifikasi yang bercabang dari bidang ilmu data mining. Penelitian ini menggunakan algoritma *K-Nearest Neighbor* atau yang biasa disebut dengan *kNN* pada proses pengenalan aktivitas manusia. Algoritma *kNN* mengklasifikasikan objek atau data baru berdasarkan atribut dan training sample. Algoritma *kNN* mengukur jarak antara data baru dengan training sample dengan menggunakan *Euclidean Distance*.

Melihat permasalahan kecelakaan lalu lintas yang telah dipaparkan sebelumnya, maka penelitian ini berusaha memberikan solusi untuk mengurangi kecelakaan lalu lintas yang disebabkan oleh penggunaan telepon pintar ketika berkendara. Adapun solusi yang ditawarkan yaitu dengan memanfaatkan sensor *accelerometer* pada handphone. Sensor *accelerometer* yang digunakan untuk mendeteksi aktivitas berkendara, serta sebuah aplikasi untuk membalas pesan secara otomatis, maka diharapkan mampu menciptakan kondisi berkendara yang aman.

II. LANDASAN TEORI

A. Aktivitas Manusia

Aktivitas manusia diwujudkan melalui suatu tindakan dari individu-individu yang berpartisipasi di dalamnya. Aktivitas dapat dikelompokkan ke dalam beberapa jenis. Pertama, ada aktivitas sederhana yang dilakukan manusia seperti; berjalan, jogging, menuruni tangga, mengambil lift, dan lain-lain. Kedua, ada aktivitas kompleks yang biasanya mengombinasikan aktivitas menjadi aktivitas dengan waktu yang lama, seperti: menaiki bus, mengemudi, dan lain-lain. Lalu yang ketiga terdapat aktivitas yang dapat dilakukan hanya dengan beberapa anggota tubuh seperti; mengetik, melambaikan tangan, dan lain-lain [2].

Aktivitas manusia juga dapat dikenali menggunakan sensor seperti; Accelerometer, Gyroscope, Camera, dan GPS. Smartphone merupakan perangkat yang kecil dan mudah dibawa kemana-mana. Pada penelitian ini smartphone digunakan untuk mengumpulkan data sensor *accelerometer*. Performa *smartphone* saat ini juga sudah cukup untuk melakukan penelitian ini.

B. Sensor Accelerometer

Sensor *Accelerometer* merasakan percepatan akibat gravitasi bumi. Pembacaan sensor accelerometer termasuk pembacaan tiga sumbu dari arah yang berbeda [4]. Sensor accelerometer telah dimanfaatkan diberbagai bidang, seperti: transportasi, medis, *science and engineering*, dan perangkat elektronik.

C. K-Nearest Neighbor

Algoritma *K-Nearest Neighbor* atau yang biasa disebut KNN pertama kali dikenalkan pada awal tahun 1950-an. Metode ini bekerja sangat baik dengan data *training* yang besar, metode ini belum begitu dikenal hingga tahun 1960 dimana teknologi komputer semakin berkembang. Metode ini banyak digunakan di bidang pengenalan pola [8].

Cara kerja algoritma KNN sangat sederhana, yaitu bekerja berdasarkan jarak terpendek/terdekat dari *query instance* ke *training sample* untuk menentukan KNN-nya. Pada penelitian ini, dekat atau jauhnya tetangga dihitung berdasarkan *Euclidean Distance*. *Euclidean distance* memiliki fungsi untuk menguji ukuran yang bisa digunakan sebagai interpretasi kedekatan jarak antara dua obyek. Adapun rumus untuk menghitung jarak *euclidean distance* dapat dilihat dibawah ini.

$$\sqrt{\sum_{i=1}^k (x_i - y_i)^2} \quad (1)$$

Pada rumus diatas, x merupakan atribut dari data *training*, sedangkan y merupakan atribut dari data yang baru.

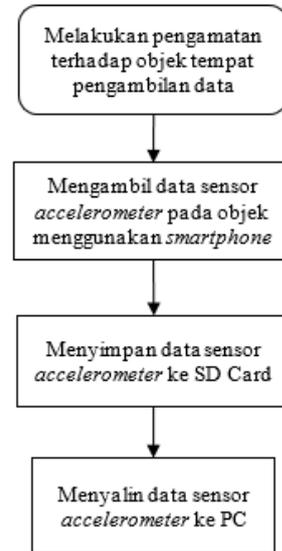
D. Android

Android merupakan sebuah sistem operasi perangkat *mobile* berbasis Linux yang mencakup sistem operasi, *middleware*, dan aplikasi [7]. Pada penelitian ini, aplikasi pengambilan data sensor *accelerometer* dan aplikasi *auto feedback* akan dikembangkan pada sistem operasi Android. Sistem operasi Android dipilih sebagai platform untuk aplikasi yang akan dikembangkan karena sistem operasi Android yang *open source*, dukungan komunitas yang banyak, serta penggunaanya yang banyak dapat membuat aplikasi dapat digunakan oleh banyak orang.

III. METODOLOGI PENELITIAN

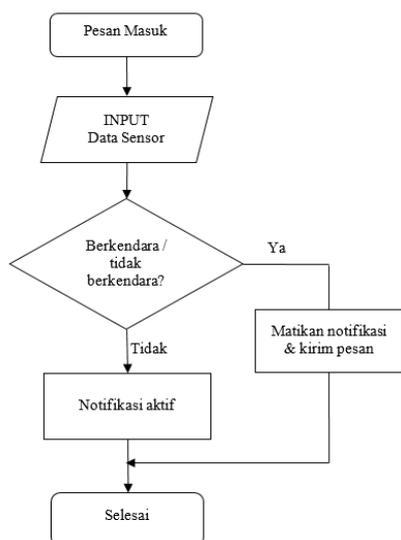
A. Perancangan Sistem

Berikut ini adalah tahapan-tahapan penelitian secara garis besar yang disajikan dalam bentuk *flowchart* yang dapat dilihat pada gambar 1 di bawah ini:



Gambar 1. Flowchart Tahapan-Tahapan Penelitian

Tahap pertama adalah mengumpulkan data *accelerometer* menggunakan *smartphone*. Data yang telah dikumpulkan kemudian dihilangkan *noise*-nya. Setelah itu lakukan klasifikasi aktivitas berdasarkan data yang telah dihilangkan *noise*-nya menggunakan algoritma KNN. Aktivitas yang telah dilasifikasikan kemudian digunakan untuk membuat aplikasi *auto feedback* pada Android. Aplikasi *auto feedback message* berfungsi untuk membalas sms secara otomatis ketika pengguna *handphone* sedang berkendara.



Gambar 2: Flowchart aplikasi Auto Feedback Android

Berikut ini adalah penjelasan flowchart di atas. Ketika ada pesan masuk ke *handphone*, maka aplikasi akan melakukan input data sensor *accelerometer handphone* tersebut. Setelah itu data sensor *accelerometer* akan dibandingkan dengan data aktivitas berkendara yang telah dibuat. Jika data sensor *accelerometer* termasuk ke dalam kategori “Berkendara”, maka notifikasi pesan akan dimatikan dan akan mengirimkan pesan secara otomatis ke pengirim pesan. Namun Jika data sensor *accelerometer* termasuk ke dalam kategori “Tidak Berkendara”, maka notifikasi pesan akan aktif dan pesan akan diterima.

B. Pengumpulan Data

Pada penelitian ini, teknik pengumpulan data yang digunakan adalah kualitatif, yaitu dengan melakukan observasi secara langsung. Data yang diambil berupa data sensor *accelerometer* yang berasal dari *smartphone*. Pada aktivitas berkendara, *smartphone* diletakkan pada setir mobil dan stang motor. Sedangkan pada aktivitas tidak berkendara, *smartphone* diletakkan di saku depan celana. Pengambilan data dilakukan dengan menggunakan program *Accelerometer Capturer* yang dijalankan pada *smartphone* Android.

Pada aktivitas berkendara, pengambilan data dilakukan dengan melakukan perjalanan menggunakan mobil dan motor dengan rute yang sama dengan kisaran waktu ± 30 menit. Kemudian untuk aktivitas tidak berkendara, pengambilan data dilakukan dengan melakukan aktivitas sederhana seperti: duduk, berjalan, dan naik & turun tangga. Data sensor *Accelerometer* akan secara otomatis tersimpan di ruang penyimpanan *smartphone* dengan format **.txt*. Data tersebut terdiri dari: koordinat *x*, koordinat *y*, koordinat *z*, dan waktu pengambilan data.

C. Pengolahan Data

Pada penelitian ini, ada 3 proses yang dilakukan untuk mengolah data sensor *Accelerometer*. Pengolahan data

dilakukan dengan menggunakan *software* Microsoft Excel. Berikut ini adalah proses-proses yang dilakukan pada pengolahan data.

1) *Kombinasi Data*: Kombinasi data merupakan proses mengkombinasikan koordinat *x*, *y*, dan *z* pada data sensor *accelerometer*. tujuan dari proses ini adalah untuk memudahkan dalam mencari baris data koordinat *x*, *y*, dan *z* yang memiliki duplikat.

2) *Mencari Data Duplikat*: Pencarian data duplikat dilakukan dengan menggunakan formula IF dan COUNTIF pada Microsoft Excel. Data yang dihasilkan dari proses Kombinasi Data yang telah dilakukan sebelumnya akan dijadikan sebagai ukuran untuk mencari data yang duplikat. Berikut ini adalah formula IF dan COUNTIF yang digunakan.

3) *Acak Data*: Data pada masing-masing aktivitas tidak akan digunakan seluruhnya. Oleh karena itu masing-masing aktivitas akan diambil beberapa datanya secara acak. Untuk mengambil data secara acak, formula yang digunakan adalah =RAND() yang akan menghasilkan angka acak. Angka acak tersebut selanjutnya diurutkan secara ascending.

D. Penyajian Data

Setelah data dari beberapa jenis aktivitas dijadikan satu kesatuan, data tersebut kemudian dibagi menjadi data training sebanyak 80% dan data testing sebanyak 20%. Presentase dari data training lebih banyak dikarenakan algoritma KNN tangguh terhadap data training yang banyak. Pada pembagian data training dan data testing, data dikelompokkan menjadi 2 kelas, yaitu kelas “Berkendara” dan kelas “Tidak Berkendara”. Untuk data berkendara mobil dan berkendara motor akan masuk ke dalam kelas “Berkendara”. Kemudian untuk data duduk di kursi, duduk sila, berjalan, naik dan turun tangga akan masuk ke dalam kelas “Tidak Berkendara”.

IV. HASIL DAN PEMBAHASAN

A. Hasil Implementasi

Hasil implementasi pada penelitian ini merupakan sebuah aplikasi *auto feedback* yang berjalan pada sistem operasi Android. Aplikasi *auto feedback* akan melakukan klasifikasi aktivitas berkendara secara terus menerus menggunakan algoritma KNN. Data yang dikalkulasi menggunakan algoritma KNN disimpan pada sebuah *database*. Aplikasi *auto feedback* akan merubah profil *smartphone* menjadi *silent mode* dan membalas pesan secara otomatis apabila mendeteksi pengguna *smartphone* sedang berkendara. Namun, apabila pengguna sedang tidak berkendara, maka pesan akan diterima seperti biasa.

1) Implementasi Algoritma K-Nearest Neighbor:

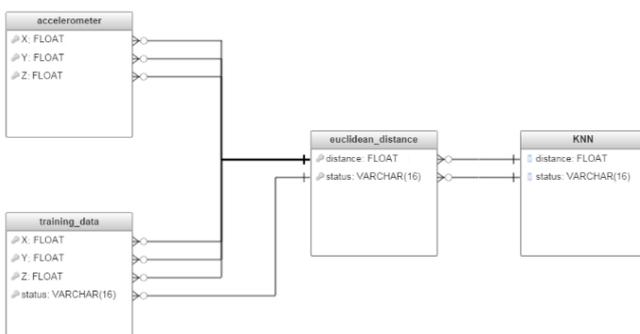
Implementasi algoritma KNN pada penelitian ini diterapkan pada *data set* yang telah dipersiapkan sebelumnya pada Microsoft Excel. Berikut ini adalah langkah-langkah implementasi algoritma KNN pada penelitian ini:

- Menentukan nilai k pada algoritma KNN. Nilai k yang digunakan adalah $k = 7$.
- Menghitung jarak *euclidean distance* antara sebuah data *testing* dengan seluruh data *training*.
- Mengurutkan jarak *euclidean distance* secara *ascending*.
- Menghitung jumlah kelas terbanyak dari 7 jarak *euclidean distance* teratas karena pada penelitian ini nilai k yang digunakan pada algoritma KNN adalah $k = 7$.
- Jumlah kelas terbanyak menentukan kelas dari data yang baru. Jika jumlah kelas “Berkendara” > 3 , maka data yang baru akan diklasifikasikan sebagai kelas “Berkendara”. Namun, jika jumlah kelas “Tidak Berkendara” > 3 , maka data yang baru akan diklasifikasikan sebagai kelas “Tidak Berkendara”.

Dari seluruh percobaan yang dilakukan sebanyak jumlah data *testing*, ketepatan algoritma KNN dalam melakukan klasifikasi aktivitas berkendara adalah sebesar 87.88%.

2) Database

Dari seluruh percobaan yang dilakukan sebanyak jumlah data *testing*, ketepatan algoritma KNN dalam melakukan klasifikasi aktivitas berkendara adalah sebesar 87.88%.



Gambar 3. Skema Database

Berikut ini adalah penjelasan tabel dan *view* yang terdapat pada skema *database* pada gambar 3 di atas.

- **Tabel accelerometer**
Tabel accelerometer pada *database* digunakan untuk menyimpan data *accelerometer* yang baru. Data pada tabel *accelerometer* hanya terdiri satu baris saja. Ketika aplikasi *auto feedback* dijalankan, aplikasi hanya akan melakukan *update* data pada tabel ini.
- **Tabel training_data**
Tabel *training_data* pada *database* yang digunakan untuk menyimpan data *training* yang sebelumnya telah diuji.
- **View euclidean_distance**
View euclidean_distance merupakan *view* yang berisi data jarak *euclidean distance* yang berasal dari perhitungan antara tabel *accelerometer* dengan tabel *training_data*.

- **View KNN**
View KNN merupakan *view* yang berisi data sebanyak 7 baris yang berasal dari *query view euclidean_distance* yang diurutkan secara *ascending*. Data yang diambil hanya 7 baris karena nilai k yang ditentukan untuk algoritma KNN pada penelitian ini adalah 7.

3) Aplikasi Auto Feedback

Aplikasi *auto feedback* berjalan pada *background service*, sehingga aplikasi *auto feedback* akan tetap berjalan meskipun pengguna sedang menjalankan aplikasi lain. Berikut ini adalah *user interface* aplikasi *auto feedback* yang dapat dilihat pada gambar 4 di bawah ini.



Gambar 4. User Interface versi Auto Feedback

Berikut ini adalah penjelasan dari *user interface* aplikasi *auto feedback* pada gambar 4 di atas.

- **Tombol Start**
Tombol *start* merupakan tombol yang digunakan untuk menjalankan aplikasi *auto feedback*. Ketika pengguna membuka aplikasi *auto feedback*, pengguna harus terlebih dahulu menekan tombol ini agar aplikasi *auto feedback* dapat berjalan pada *background service*.
- **Tombol Stop**
Ketika pengguna telah menekan tombol *start*, maka secara otomatis tombol *stop* akan muncul. Tombol *Stop* digunakan untuk menghentikan *service* yang berjalan pada *background*.
- **Tombol Exit**
Tombol *exit* digunakan untuk keluar dari aplikasi. Ketika program sedang berjalan pada *background service* namun pengguna menekan tombol *exit*, maka *service* yang sedang berjalan pada *background* akan dihentikan secara otomatis.
- **Koordinat Accelerometer**
Koordinat *accelerometer* menampilkan perubahan dari koordinat sensor *accelerometer* ketika *smartphone* digerakkan. Terdapat ± 5 kali perubahan koordinat *accelerometer* dalam 1 detik.
- **Feedback Count**

Feedback *count* menampilkan jumlah *feedback* pesan yang dikirim ketika pengguna sedang berkendara.

B. Pengujian Sistem

Pengujian yang dilakukan pada penelitian ini merupakan pengujian aplikasi auto feedback untuk mengukur seberapa tepat aplikasi tersebut dapat mengenali aktivitas berkendara dan aktivitas tidak berkendara. Peralatan yang digunakan pada pengujian sistem yaitu: *car smartphone holder*, *motorcycle smartphone holder*, dan 2 buah *smartphone* yang berfungsi sebagai pengirim dan penerima pesan.

1) Pengujian Aktivitas Berkendara

Pengujian aktivitas berkendara dibagi menjadi 2, yaitu pengujian berkendara motor dan pengujian berkendara mobil. Pengujian aktivitas berkendara dilakukan dengan melakukan perjalanan dari satu tempat ke tempat lain dengan menggunakan sepeda motor dan mobil. Pada pengujian aktivitas berkendara motor, *smartphone* yang berfungsi sebagai penerima pesan akan menjalankan aplikasi *auto feedback* dan diletakkan pada stang motor menggunakan perangkat *motorcycle smartphone holder*.



Gambar 5. Pengujian aplikasi auto feedback pada aktivitas berkendara motor

Pada pengujian aktivitas berkendara mobil, *smartphone* yang berfungsi sebagai penerima pesan akan menjalankan aplikasi *auto feedback* dan diletakkan pada setir mobil menggunakan perangkat *car smartphone holder*.



Gambar 6. Pengujian aplikasi auto feedback pada aktivitas berkendara mobil

Pengujian aktivitas berkendara dilakukan dengan mengirimkan pesan menggunakan *smartphone* yang berfungsi sebagai pengirim pesan ke *smartphone* penerima pesan yang menjalankan aplikasi *auto feedback* pada saat pengguna *smartphone* sedang mengendarai motor atau mobil. Banyaknya pesan yang dikirim pada masing-masing aktivitas

berkendara sebanyak 20 pesan, sehingga total pesan yang dikirim adalah sebanyak 40 pesan.

2) Pengujian Aktivitas Tidak Berkendara

Pada pengujian aktivitas tidak berkendara, *smartphone* yang berfungsi sebagai penerima pesan akan menjalankan aplikasi *auto feedback* dan diletakkan pada saku depan celana. Kemudian *smartphone* yang berfungsi sebagai pengirim pesan akan mengirimkan pesan ke *smartphone* yang sedang menjalankan aplikasi *auto feedback*.

Pengujian dilakukan dengan mengirimkan pesan ke *smartphone* yang menjalankan aplikasi *auto feedback* pada saat pengguna *smartphone* sedang melakukan aktivitas sederhana seperti duduk sila, duduk di kursi, berjalan, dan naik & turun tangga. Banyaknya pesan yang dikirim pada masing-masing aktivitas tersebut sebanyak 10 pesan, sehingga total pesan yang dikirim adalah sebanyak 40 pesan.

C. Pembahasan

Pada pengujian aktivitas berkendara, aplikasi dianggap berhasil mengenali aktivitas berkendara apabila pesan diterima dengan *silent mode* dan akan mengirimkan *feedback* pesan ke *smartphone* yang mengirimkan pesan. Berikut ini adalah hasil pengujian aktivitas berkendara yang dapat dilihat pada tabel 1 di bawah ini.

TABEL 1. HASIL PENGUJIAN APLIKASI *AUTO FEEDBACK* PADA AKTIVITAS BERKENDARA

Aktivitas	Pesan yang Mendapat Feedback	Pesan yang Tidak Mendapat Feedback	Total Pesan
Berkendara Motor	17 (85%)	3 (15%)	20 Pesan
Berkendara Mobil	19 (95%)	1 (5%)	20 Pesan
Total	36 (90%)	4 (10%)	40 Pesan

Berdasarkan hasil pengujian pada tabel 1 di atas, ketepatan aplikasi *auto feedback* untuk mengenali aktivitas berkendara sudah baik dimana presentase ketepatan aplikasi untuk mengenali aktivitas berkendara mencapai 90%. Pada pengujian aktivitas tidak berkendara, aplikasi dianggap berhasil mengenali aktivitas apabila pesan diterima secara normal dan tidak mengirim *feedback* berupa pesan ke pengirim pesan. Berikut ini adalah hasil pengujian aktivitas tidak berkendara yang dapat dilihat pada tabel 2 di bawah ini.

TABEL 2. HASIL PENGUJIAN APLIKASI *AUTO FEEDBACK* PADA AKTIVITAS TIDAK BERKENDARA

Aktivitas	Pesan yang Mendapat Feedback	Pesan yang Tidak Mendapat Feedback	Total Pesan
Berjalan	7 (70%)	3 (7.5%)	10 Pesan
Naik dan turun tangga	8 (80%)	2 (5%)	10 Pesan

Duduk di kursi	9 (90%)	1 (2.5%)	10 Pesan
Duduk sila	10 (100%)	0 (0%)	10 Pesan
Total	34 (85%)	6 (15%)	40 Pesan

Berdasarkan hasil pengujian pada tabel 2 di atas, ketepatan aplikasi *auto feedback* untuk mengenali aktivitas tidak berkendara sudah baik. Hal ini dapat dilihat dari pesan yang tidak mendapatkan *feedback* mencapai 85% dari total keseluruhan pesan yang dikirim yang menandakan bahwa aplikasi mendeteksi bahwa pengguna *smartphone* sedang tidak berkendara. Berdasarkan data pada tabel 2 di atas, pesan yang mendapat *feedback* yang merupakan kesalahan dalam pengenalan aktivitas tidak berkendara terdapat pada aktivitas berjalan dan naik & turun tangga. Hal ini mungkin dikarenakan pada kedua aktivitas tersebut merupakan aktivitas yang lebih banyak melakukan pergerakan dibandingkan dengan aktivitas duduk sila dan duduk di kursi.

V. KESIMPULAN DAN SARAN

Berdasarkan penelitian dan pengujian yang telah dilakukan, maka dapat disimpulkan bahwa penerapan algoritma K-Nearest Neighbor dalam pengenalan aktivitas berkendara sudah baik. Hal ini dapat dilihat pada hasil pengujian aplikasi *auto feedback* yang dapat mengenali aktivitas sedang berkendara dengan ketepatan hingga 95% dan aktivitas tidak berkendara dengan ketepatan 85%.

Mengingat masih terdapat kekurangan pada penelitian ini, maka penulis memberikan beberapa saran yang diharapkan dapat bermanfaat untuk pengembangan dari penelitian ini.

- 1) Agar dapat melakukan klasifikasi lebih tepat dan akurat, dapat ditambahkan sensor seperti GPS yang dapat mengukur kecepatan.
- 2) Feedback dikirim tidak hanya sebatas ketika terdapat pesan masuk saja.

- 3) Menjaga layar *smartphone* tetap dalam kondisi mati ketika terdapat pesan masuk, karena saat ini beberapa *smartphone* masih menyalakan layar ketika terdapat pesan masuk meskipun pesan diterima dengan silent mode.
- 4) Aplikasi *auto feedback* dapat dijalankan dengan sempurna pada seluruh versi sistem operasi Android, karena saat ini aplikasi *auto feedback* hanya dapat berjalan dengan sempurna pada Android versi 4.4.4 Kitkat.
- 5) Menambah kompatibilitas pada aplikasi *auto feedback* sehingga dapat berjalan pada sistem operasi mobile lainnya, seperti iOS, Windows Phone, dan lain-lain.

DAFTAR PUSTAKA

- [1] World Health Organization. 2016. Global Status Report On Road Safety 2015. Italy: World Health Organization, pp. 38-43.
- [2] Karpatschhof, B. 2000. Human activity, Contribution to the Anthropological Sciences from a Perspective of Activity Theory. Copenhagen: Dansk Psykologisk Forlag, p. 184.
- [3] Vacek, S., Steffen, K., & Rudiger, D. Classifying Human Activities in Household Environments. pp. 2-3.
- [4] Su, X. 2014. Activity Recognition with Smartphone Sensors Second Exam Literature Review. pp. 5-6.
- [5] Kaghyan, S. and Sarukhanyan, H. 2012. Activity Recognition Using K-Nearest Neighbor Algorithm on Smartphone with Tri-Axial Accelerometer. International Journal "Information Models and Analyses" Vol.1 / 2012, pp. 150-155.
- [6] Kose, M., Durmaz, O., & Ersoy, C. 2012. Online Human Activity Recognition on Smart Phones. 2nd International Workshop on Mobile Sensing, Beijing, China, April 16, 2012. Pp. 1-5.
- [7] Kurniawan, D.E., 2017. Push Notification System Pada Prototipe Kendali Listrik Rumah. CESS (Journal of Computer Engineering, System and Science), 2(2), pp.89-92.
- [8] Supardi, Y. 2011. Semua Bisa Menjadi Programmer Android Basic. Jakarta: PT Elex Media Komputindo, pp. 1-6.
- [9] Han, J. and Kamber, M. 2006. Data Mining Concepts and Techniques. United State of America: Diane Cerra, pp. 148-156.
- [10] Kurniawan, D.E. and Surur, M.N., 2016. Perancangan Sistem Pengamanan Sepeda Motor Menggunakan Mikrokontroler Raspberry Pi dan Smartphone Android. Jurnal Komputer Terapan, 2(2), pp.93-104.