

ANALISIS PERFORMANSI METODE KNN (K-NEAREST NEIGHBOR) UNTUK PENGENALAN KARAKTER PADA PLAT NOMOR KENDARAAN DI RASPBERRY PI

PERFORMANCE ANALYSIS OF KNN (K-NEAREST NEIGHBOR) METHOD FOR LICENSE PLATE CHARACTER RECOGNITION IN RASPBERRY PI

Noor Azlin Tauchid¹, R Rumani², Budhi Irawan, S.Si., M.T.³

^{1,2,3}Fakultas Teknik Elektro – Universitas Telkom

Jln. Telekomunikasi No.1 Terusan Buah Batu Bandung 40257 Indonesia

¹caintheholyydragoon@students.telkomuniversity.ac.id, ²rumani@telkomuniversity.ac.id,

³bir@telkomuniversity.ac.id

Abstrak

Proses parkir adalah proses yang sangat penting pada dunia yang terus bergerak. Dengan berkembangnya dunia otomotif, jumlah pengguna kendaraan terus meningkat. Keberadaan tempat parkir sangat diperlukan dan sistem yang cepat dan otomatis. Komputer dengan kemampuannya, selalu didorong untuk memenuhi kemampuannya dalam membantu tugas manusia. Terutama karena memiliki kelebihan dalam hal komputasi dan efisiensi. Dalam hal ini, telah dibuat sistem yang mampu mengenali karakter pada plat nomor sehingga mampu membantu dalam proses pencatatan. Sistem yang dibuat menggunakan metode KNN. KNN pada aplikasi yang dibuat mampu mendeteksi plat nomor dengan tingkat akurasi sebesar 77.98% dan waktu deteksi rata-rata sebesar 9645.988133 ms ketika dijalankan di Raspberry Pi tipe B.

Kata Kunci : KNN, OCR, Raspberry Pi

Abstract

Parking process is one of vital things in our running world. With the growth of vehicle industry, the number of vehicle user is also growing rapidly. So the needs for parking place and fast and automatic parking system is very anticipated. With Computer myriad abilities, Computer are always pushed through their limit to meet the human prospect. Especially in computational and efficiency abilities. With that problems in mind, systems to recognizing character in license plate are made, so it can help in registering process. Systems are made with KNN method. KNN in made application capable to detect character in image of license plate with accuracy of 77.98% and 9645.988133 ms average detection time in Raspberry Pi Type B.

Keyword : KNN, OCR, Raspberry Pi.

1. Pendahuluan

Parkir adalah kegiatan manusia sehari-hari. Setiap pemilik kendaraan haruslah memarkir kendaraannya jika dibawa dalam beraktifitas. Dalam hal ini dikebanyakan tempat parkir, kebanyakan manusia masih mengambil alih dalam tugas melakukan pencatatan dan identifikasi kendaraan saat memasuki lahan parkir.

Jika dilihat dari jumlah petugas parkir yang umum dari suatu lahan parkir, biasanya untuk parkir skala kecil menggunakan 2-3 tenaga manusia dalam pengoperasiannya. Untuk skala yang lebih besar, jumlah petugas parkir yang dibutuhkan akan lebih banyak lagi. Tentunya itu akan membuat anggaran untuk membayar pekerja menjadi membengkak dan kurang praktis.

Sebenarnya peran manusia dalam mengerjakan pekerjaan pencatatan dapat dikurangi. Hal ini dilakukan dengan memasukan mesin sebagai pengganti tugas manusia dalam mengerjakan pekerjaan pencatatan. Dengan mengganti mesin sebagai petugas pencatatan, diharapkan efisiensi dan optimalisasi dapat didapat karena sifat mesin yang presisi.

Permasalahan inilah yang menjadi fokus utama dari ditulisnya Tugas Akhir ini. Perbandingan antara 2 algoritma yang digunakan, diharapkan dapat diketahui perbedaan performansi antara kedua metode dalam sistem untuk menemukan mana yang lebih baik diantara keduanya dari parameter yang diujikan..

KNN telah sering digunakan dalam pendeteksian karakter, termasuk pendeteksian tulisan tangan. KNN dapat diaplikasikan dengan mudah karena untuk memperoleh hasil deteksi, klasifikasi yang perlu dilakukan hanyalah dengan memberi label pada data training yang akan digunakan sebagai komparasi pada sistem deteksi. Tujuan

utama yang diinginkan adalah mengetahui performansi KNN sebagai *lazy machine learning* pada proses deteksi plat nomor menggunakan Raspberry Pi.

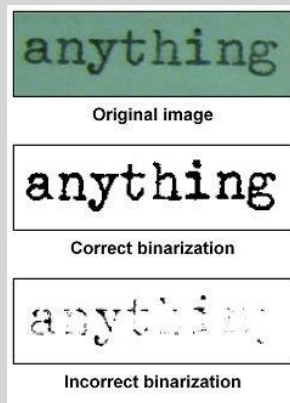
2. Dasar Teori

2.1 Plat Nomor Kendaraan

Plat nomor yang diujikan hanyalah plat nomor dengan spesifikasi yang legal, terutama pada daerah Jawa Barat. Berdasarkan surat Kapolri nomor: B/588/v/2011 tertanggal 23 Mei 2011 tentang pengiriman perubahan spesifikasi teknis materiil surat tanda nomor kendaraan (STNK) 2012 dan tanda nomor kendaraan bermotor (TNKB).

2.2 OCR (Optical Character Recognition)

OCR secara definisi adalah teknologi yang digunakan untuk mengonversi gambar dengan konten teks didalamnya menjadi teks yang dapat diedit atau disimpan dalam format teks seperti TXT, DOC atau PDF. Penggunaan OCR sudah banyak digunakan secara umum, contohnya pada konversi gambar ke teks pada hasil pindai buku tulis.



Gambar 2.1: Contoh gambar yang mengalami *binarization* sebagai salah satu contoh proses OCR OpenCV

menyediakan beberapa metode OCR pada website opencv yang bebas diunduh dan dipergunakan. Metode OCR yang digunakan adalah OCR yang berbasis KNN.

2.3 K-Nearest N (KNN)

K-Nearest Neighbour adalah algoritma yang berbasis ide bahwa pola terdekat dengan pola target x^1 memiliki informasi yang berguna. KNN menetapkan tipe *class label* pada kebanyakan pola K-nearest pada data space. Untuk dapat dimanfaatkan, kemiripan pola pada data space harus dapat didefinisikan terlebih dahulu.

KNN classifier merupakan ekstensi dari yang sistem klasifikasi Nearest Neighbor awal (NN). KNN classifier berjalan berdasarkan keputusan nonparametrik sederhana. Setiap Iq citra query diperiksa berdasarkan jarak fitur-fiturnya dari fitur gambar lain dalam *training* database. Titik K terdekat adalah gambar yang memiliki jarak minimum dari citra query dalam ruang fitur. Jarak antara dua fitur dapat diukur berdasarkan salah satu fungsi jarak seperti, city-block distance d_1 (2.1), dan d_2 jarak Euclidean (2.2) atau jarak cosinus d (2.3).

$$d_1(x, y) = \sum_{i=1}^N |x_i - y_i| \quad \dots (2.1)$$

$$d_2(x, y) = \sqrt{\sum_{i=1}^N |x_i - y_i|^2} \quad \dots (2.2)$$

$$d_3(x, y) = 1 - \frac{\vec{x} \cdot \vec{y}}{|\vec{x}| |\vec{y}|} \quad \dots (2.3)$$

2.4 Bahasa Pemrograman Python

Python merupakan bahasa pemrograman yang freeware atau perangkat bebas dalam arti sebenarnya, tidak ada batasan dalam penyalinannya atau mendistribusikannya. Lengkap dengan source code-nya, debugger dan profiler, antarmuka yang terkandung di dalamnya untuk pelayanan antarmuka, fungsi sistem, GUI (antarmuka pengguna grafis), dan basis datanya. Python menjadi bahasa official yang terintegrasi dalam Raspberry Pi. Kata “Pi” pada Raspberry Pi merupakan slang yang merujuk pada “Python”.

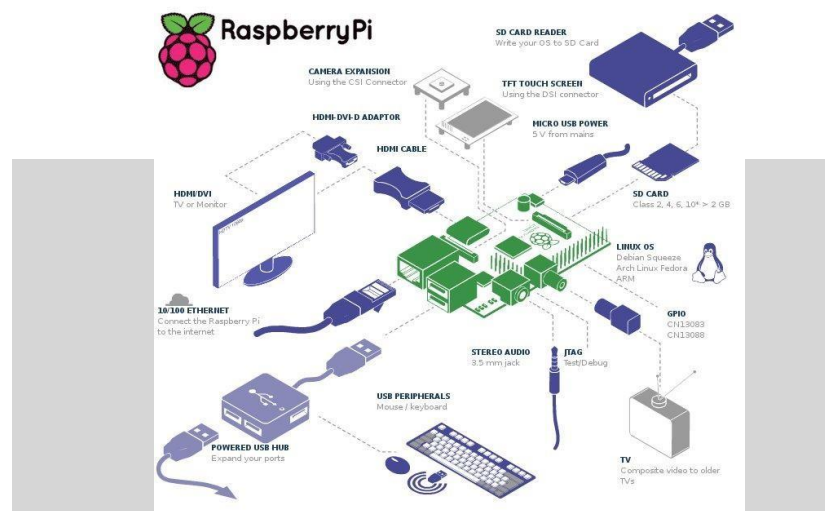
2.5 OpenCV

OpenCV (Open Source Computer Vision Library) adalah *open source* pengolahan citra pada komputer dan *machine-learning library*. OpenCV dibangun untuk menyediakan infrastruktur umum untuk aplikasi komputer dan untuk mempercepat penggunaan kecerdasan buatan pada komputer dalam produk komersial. Menjadi produk BSD-lisensi, OpenCV memudahkan bagi perusahaan untuk memanfaatkan dan memodifikasi kode.

Perpustakaan memiliki lebih dari 2.500 algoritma yang dioptimalkan, mencakup seperangkat baik klasik maupun *state-the-art* pengolahan citra dan algoritma untuk *machine-learning*. Algoritma ini dapat digunakan untuk mendeteksi dan mengenali wajah, mengidentifikasi objek, mengklasifikasikan tindakan manusia dalam video, gerakan track kamera, track objek bergerak, ekstrak model 3D dari objek, menghasilkan titik 3D dari kamera stereo di cloud, menggabungkan gambar bersama-sama untuk menghasilkan gambar beresolusi tinggi, menemukan gambar yang sama pada database gambar, menghilangkan *red-eyes* dari gambar yang diambil dengan menggunakan flash, mengikuti gerakan mata, mengenali pemandangan dan menetapkan penanda untuk overlay dengan augmented reality, dll OpenCV memiliki lebih dari 47 ribu orang pengguna masyarakat dan diperkirakan jumlah download melebihi 7 juta. Perpustakaan digunakan secara luas dalam perusahaan, kelompok penelitian dan oleh badan-badan pemerintah.

2.6 Raspberry Pi

Raspberry Pi adalah komputer berukuran kecil sebesar kartu kredit. Raspberry Pi memiliki fungsi serupa dengan komputer pada umumnya. Ada bagian USB untuk memasukkan keyboard dan mouse. Terdapat 2 model pada Raspberry Pi, model A dan model B. Beberapa perbedaan di antara keduanya terletak pada ukuran memori (256 MB pada model A dan 512 MB pada model B) serta ketersediaan network adaptor yang hanya ada pada model B.



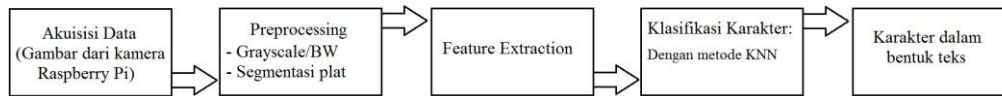
Gambar 2.2: Arsitektur Raspberry Pi

3. Pembahasan

3.1 Spesifikasi Perancangan dan Prakira

Sistem yang dibuat adalah sistem deteksi objek pada gambar plat nomor yang diambil dalam proses parkir. Objek yang dideteksi adalah plat nomor dan karakter pada gambar objek dideteksi oleh OCR menggunakan metode KNN.

Sistem yang dibuat terdiri dari 3 bagian, yaitu pengambilan gambar dengan kamera yang dipicu dengan sensor, deteksi plat nomor, deteksi karakter menggunakan metode OCR yang ingin diperbandingkan dan perbandingan performansi dari 2 metode yang digunakan. Skema sistem tersebut dapat digambarkan sebagai berikut:



Gambar 3.1: Blok diagram dari sistem yang akan dibuat

3.2 Cara pengujian

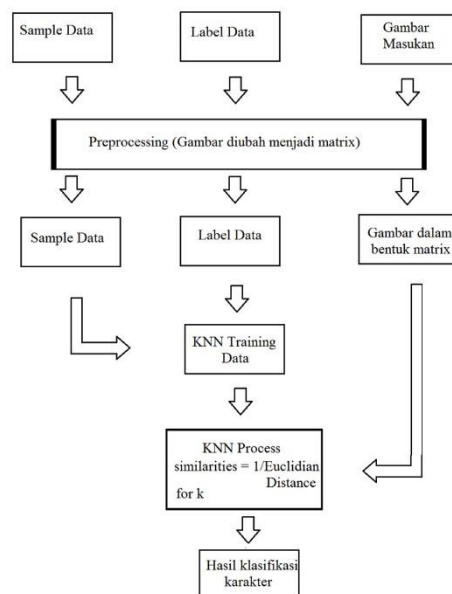
Pada proses deteksi plat nomor kendaraan, akan digunakan fungsi *contour* pada OpenCV. Seperti yang telah disebutkan sebelumnya, proses deteksi yang merupakan proses yang terdiri dari proses mengambil masukan, proses preprocessing dan ekstraksi ciri. Masukan didapat menggunakan kamera pada Raspberry Pi yang diambil pada jarak 50 cm. Gambar mobil dengan plat nomor sebelumnya akan mengalami preprocessing sebelum mengalami proses ekstraksi ciri. Preprocessing yang dilakukan antara lain sebagai berikut:

1. Konversi gambar menjadi grayscale
2. Konversi gambar menjadi BW dengan menggunakan proses thresholding
3. Proses segmentasi untuk men-crop plat nomor menggunakan fitur *contour*.

Sementara itu proses ekstraksi ciri yang dilakukan antara lain:

1. Deteksi tepi dan dilasi untuk memperjelas karakter
2. Proses memisahkan karakter dengan fitur bounding contour
3. Mengubah gambar menjadi array float32

Deteksi karakter menggunakan KNN



Gambar 3.2: Blok diagram cara kerja metode KNN dalam proses klasifikasi

Training data yang digunakan adalah training data yang didapatkan dalam proses training sebelumnya. Training data terdiri dari 2 training data: sample dan label. Dari data yang didapatkan diatas, sample didapatkan dari sample karakter yang dideteksi menggunakan training machine. Sementara label didapatkan dari variable

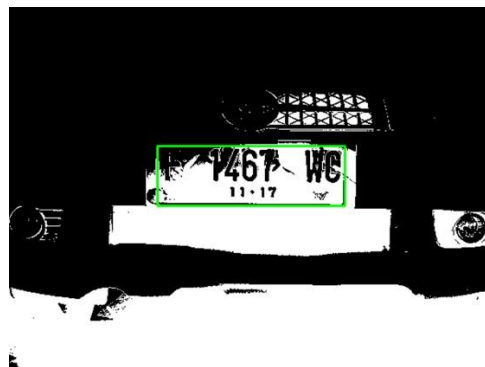
char yang diinputkan sesuai ASCII.

Proses klasifikasinya akan menggunakan library yang dimiliki OpenCV, yaitu `CV2.Knearest()`. Fungsi ini berfungsi dengan mendeteksi jarak kemiripan terdekat dari object yang akan diklasifikasi dengan sample yang ada menggunakan Euclidian Distance sejumlah K . K yang akan digunakan adalah $K = 5$.

3.3 Hasil Pengujian

Pengujian Program yang Menggunakan KNN

Pada pengolahan data berupa gambar dengan algoritma KNN berhasil dan dapat mendeteksi mayoritas plat nomor yang diujicobakan. Dari 30 plat nomor yang diujicobakan, ada 5 plat nomor yang tidak dapat dideteksi. Proses ekstraksi ciri dilakukan pada area objek yang berhasil di contour menggunakan fungsi `contour` pada OpenCV. Pada penggunaan gambar beresolusi 600x450 pixel, ukuran tinggi objek yang dideteksi berdasarkan nilai centroid harus lebih besar 20 pixel dan tidak lebih besar dari 50 pixel, sehingga objek yang dideteksi dapat diminimalisir hanya kepada karakter pada plat nomor saja. Nilai K yang digunakan adalah 5.



Gambar 3.3: Proses *contour* pada plat nomor kendaraan agar bisa didapatkan Region of Interest (ROI)

Setelah itu ROI yang didapatkan dari daerah yang di *contour* akan di crop sehingga menghasilkan gambar sebagai berikut:



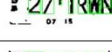
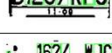
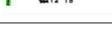


Gambar 3.4: Hasil deteksi plat nomor menggunakan fungsi *drawcontour* pada OpenCV

Setelah itu, gambar hasil dari hasil crop dibaca menggunakan program OCR yang menggunakan metode KNN. Program ini melakukan contouring kembali untuk mendapatkan karakter dan melakukan operasi OCR dengan metode KNN kepada daerah yang di contour. Daerah yang di contour disesuaikan dengan ukuran tinggi dan lebar karakter rata-rata yang diambil pada jarak 50cm.

Gambar plat adalah gambar plat setelah mengalami preprocessing. Kolom Teks pada plat adalah data factual teks plat nomor yang ada di kendaraan. Hasil deteksi adalah hasil deteksi yang didapat dari proses OCR. Warna merah pada hasil deteksi menunjukkan kesalahan dalam proses deteksi, warna biru menandakan tidak adanya karakter yang seharusnya dideteksi. Waktu deteksi adalah hitungan waktu yang didapat dari waktu preprocessing + waktu ekstraksi fitur + waktu deteksi karakter.

Tabel 3.1: Sebagian hasil deteksi menggunakan OCR dengan metode KNN

No	Gambar Plat	Teks Pada plat	Hasil Klasifikasi (Uppercase)	Waktu Deteksi
1		A1305LU	Q1305LQ	9013.131 ms
2		AE90DD	AE90DD	9743.435 ms
3		B271RWN	B271QWN	9011.26 ms
4		B1257RFS	B1257RFS	9748.425 ms
5		B1624WJC	B1624HJC	9116.68 ms

Tabel 3.2: Hasil deteksi dengan OCR berbasis metode KNN

0	1	2	3	4	5	6	7	8	9
66.67%	94.44%	92.31%	85.71%	100%	100%	50%	90.90%	66.67%	88.88%
(9)	(36)	(13)	(7)	(9)	(10)	(8)	(11)	(3)	(8)
A	B	C	D	E	F	G	H	I	J
90.90%	0%	100%	55%	50%	100%	50%	100%	0%	100%
(11)	(10)	(7)	(20)	(2)	(6)	(2)	(1)	(1)	(2)
K	L	M	N	O	P	Q	R	S	T
50%	100%	100%	80%	0%	100%	50%	80%	100%	100%
(2)	(3)	(3)	(5)	(1)	(2)	(4)	(5)	(1)	(1)
U	V	W	X	Y	Z				
100%	100%	83.33%	100%	100%	0%				
(4)	(2)	(6)	(2)	(1)	(0)				

Nilai pada kolom biru menunjukkan karakter yang di tes. Nilai dalam persen menunjukkan tingkat akurasi dari karakter tersebut saat di klasifikasi dan digit dalam kurung menunjukkan jumlah karakter secara keseluruhan. Dari test yang dilakukan, karakter “O” dan “B” adalah karakter dengan hasil deteksi paling buruk. Sementara karakter “4”, “5”, “C”, “F”, “H”, “J”, “L”, “M”, “P”, “S”, “T”, “U”, “V” dan “Y” adalah karakter yang dapat dideteksi dengan sempurna. Dari total 218 karakter yang seharusnya dapat dideteksi, ada 48 karakter yang tidak dapat terdeteksi atau salah dideteksi, sehingga akurasi aplikasi deteksi plat nomor menggunakan algoritma KNN adalah sebesar 77.98%.

4. Kesimpulan dan Saran

4.1 Kesimpulan

Dari hasil pengujian dan analisis yang telah dilakukan, kesimpulan yang dapat diambil adalah:

1. Aplikasi berbasis KNN dengan nilai $K=5$ yang difungsikan sebagai OCR mampu bekerja dengan baik dan memiliki performansi yang cukup baik. Akurasi rata-rata sebesar 77.98% dan waktu proses dibawah 10 detik menunjukkan sistem masih dapat digunakan dan diaplikasikan sesuai dengan tujuan awalnya, yaitu fungsi PARCS.
2. Deteksi yang tersulit adalah deteksi karakter “O” dan B, dikarenakan pada format plat nomor di Indonesia, karakter “O” dan “0” memiliki bentuk karakter yang sama. Sementara itu, huruf B pada plat nomor test seringkali dalam keadaan buruk dan juga seringkali menempel dengan objek lain seperti tepi pembatas sehingga tidak dapat di *contour*.
3. Hal-hal yang mempengaruhi gagalnya deteksi pada plat nomor antara lain:
 - Cat pada plat nomor sudah pudar sehingga bentuk karakter tidak jelas
 - Adanya stiker pada plat nomor yang mengganggu jalannya proses deteksi
 - Cahaya yang terlalu terang, menjadikan fungsi invert pada preprocessing menjadi noise tambahan
 - Ukuran font yang digunakan tidak baku, dapat lebih tebal atau lebih tipis, yang melewati batas toleransi dalam proses deteksi
 - Ukuran plat nomor yang tidak lazim
 - Terlalu banyak objek pada target deteksi, sehingga menyulitkan segmentasi

- Kemiringan pada plat nomor melewati batas toleransi plat nomor (15°)

1.2 Saran

Tugas akhir ini masih jauh dari sempurna dan perlu dikembangkan lebih jauh. Saran yang dapat diberikan pada tugas akhir selanjutnya adalah:

1. Penggunaan OCR menggunakan KNN sudah cukup baik, dan dataset untuk trainingnya sebaiknya menggunakan plat nomor Indonesia secara keseluruhan untuk hasil yang optimal dan didukung setidaknya sampel ribuan plat nomor dengan format yg sama untuk hasil yang maksimal.
2. Waktu deteksi yang hampir mencapai 10 detik, pada antrian yang besar dapat berpengaruh pada kenyamanan sistem parkir. Sebaiknya dilakukan *queueing* pada proses deteksi (tidak perlu real time). Penggunaan sensor ultrasonic dapat digunakan untuk mendeteksi panjang antrian dan bisa diimplementasikan langsung pada program yang akan dibuat. Dapat ditambahkan proses pemberian akses (Permit and Enforcement) dan proses pembayaran parkir yang otomatis agar dapat menjadi sistem smart parking yang lengkap.

Daftar Pustaka

- [1] Apple Inc. 2011, vImageProgramming Guide. Amerika Serikat : Penerbit Apple Inc 2011
- [2] Bayless, Steven H; Radha Neelakantan. Smart Parking and the Connected Consumer Opportunities for Facility Operators and Municipalities, diterbitkan oleh The Intelligent Transportation Society of America (ITS America), December 2012
- [3] Candrasyah, Faisal. (2014). Sistem Manajemen Daya Rumah Cerdas Berbasis Raspberry Pi. Bandung : Tugas Akhir IT Telkom.
- [4] Han, Xiaogang; Junfa Liu, Zhiqi Shen, Chunyan Miao. An Optimized K-Nearest Neighbor Algorithm for Large Scale Hierarchical Text Classification, School of Computer Engineering, Nanyang Technological University, Nanyang Ave, Singapore and School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang Ave, Singapore, diakses pada 9 April 2015
- [5] Hareendran, T.K., Arduino & Raspberry Pi Camera Interface [online], (<http://www.electroschematics.com/11140/arduino-raspberry-pi-camera-interface/>), diakses tanggal 8 April 2015)
- [6] Kaensar, Chayaporn. A Comparative Study on Handwriting Digit Recognition Classifier Using Neural Network, Support Vector Machine and K-Nearest Neighbor, Department of Mathematic, Statistic and Computer Science, Faculty of Science, Ubon Ratchathani University, Thailand, dipublikasi oleh Springer-Verlag Berlin Heidelberg 2013
- [7] Rakhman, Edi; Faisal Candrasyah, Fajar D Sutera. Raspberry Pi, Mikrokontroler Mungil Yang Serba Bisa. Yogyakarta : Penerbit Andi.
- [8] Republik Indonesia. 2012. Surat Kapolri Nomor: B/588/v/2011 Tertanggal 23 Mei 2011 Tentang Pengiriman Perubahan Spesifikasi Teknis Materiil Surat Tanda Nomor Kendaraan (STNK) 2012 Dan Tanda Nomor Kendaraan Bermotor (TNKB), BAUR TNKB SAMSAT KOTA BANDUNG, DITLANTAS POLDA JABAR.

