

Implementasi Algoritma *K-Neares Neighbor* dalam Pengenalan Aksara Jawa Berbasis Android untuk Anak Sekolah Dasar

Elkaf Rahmawan Pramudya^{1*}, Gilang Raharjito², Ajib Susanto³, Muslih⁴

^{1,2,3,4}Fakultas Ilmu Komputer, Universitas Dian Nuswantoro

* E-mail: ¹elkaf.rahmawan@dsn.dinus.ac.id, ²gilangraharjito@gmail.com,

³ajib.susanto@dsn.dinus.ac.id, ⁴muslih@dsn.dinus.ac.id

Abstrak

Teknologi saat ini berkembang semakin pesat sehingga memudahkan dunia pendidikan dalam melakukan proses belajar mengajar seperti kondisi saat ini yang sedang terkendala oleh wabah Covid-19 secara global sehingga siswa dan guru melakukan proses belajar mengajar di rumah. Berbagai aplikasi platform menawarkan proses pembelajaran online baik lewat learning management system (LMS), media sosial maupun jejaring sosial, akan tetapi beberapa materi perlu dilakukan pendekatan dan media khusus salah satu adalah materi pengenalan aksara Jawa bagi anak-anak Sekolah Dasar (SD) membutuhkan sarana latihan dan penjelasan yang menyenangkan. Penelitian ini mengusulkan aplikasi pembelajaran aksara Jawa dengan menggunakan Algoritma K-NN untuk belajar mengenal aksara Jawa dan mendeteksi kecocokan dalam penulisan aksara Jawa yang dikemas dalam aplikasi Android yang menarik dan menyenangkan. Hasil pengujian K-NN berdasarkan nilai k yang paling akurat adalah k = 5 dengan hasil berhasil 54,9% dan hasil gagal 45,1% dan diperoleh pengujian kebenaran siswa dalam mengenal aksara Jawa adalah 86%.

Kata kunci: Aksara Jawa, Android, K-NN, Pembelajaran, Sekolah Dasar

PENDAHULUAN

Teknologi saat ini berkembang semakin pesatnya sehingga memudahkan dunia pendidikan dalam melakukan proses belajar mengajar seperti kondisi saat ini yang sedang terkendala oleh wabah Covid-19 secara global sehingga siswa dan guru melakukan proses belajar mengajar di rumah. Berbagai aplikasi platform menawarkan proses pembelajaran online baik lewat *learning management system (LMS)*, media sosial maupun jejaring sosial.

Indonesia adalah Negara yang kaya berbagai macam budaya. Beranekaragam bahasa dan suku bangsa menjadi suatu kekayaan tak terhingga nilainya bagi nusa dan bangsa dari Sabang sampai Merauke. Ada 741 (Huri, 2014) bahasa daerah atau bahasa ibu yang bangsa Indonesia miliki dimana terletak antara Sabang hingga Merauke. Tak terkecuali pada teknologi *android* di Indonesia termasuk *MobileOS* nomor satu selama bulan Maret 2020 dengan *market share* Indonesia adalah 92.65% (GlobalStats, 2020), sedang pengguna *mobile phone* mencapai 338.2 juta (Kemp, 2020), perkembangan aplikasi *android* sering sekali digunakan untuk riset dan pengembangan teknologi dan aplikasi-aplikasi lainnya. Sudah banyak metode dan algoritma yang diciptakan dan digunakan untuk mempermudah kinerja

dari aplikasi *android* pada suatu objek maupun media. Berkembangnya teknologi komputer, dan banyak inovasi lagi yang telah dikembangkan oleh para peneliti, salah satunya yaitu perkembangan *android*, kemampuan *android* dalam mengenali karakter tulisan tangan aksara Jawa merupakan tantangan tersendiri dalam aplikasi di dalam *android* karena setiap bentuk karakter pada aksara Jawa terlalu rumit dan ada beberapa diantaranya memiliki struktur bentuk yang hampir sama dan memiliki kemiripan yang signifikan (Arismadhani, 2013).

Dunia pendidikan merupakan pondasi untuk membuat generasi penerus agar menjadi lebih cerdas, untuk itu generasi penerus dituntut mengikuti perkembangan Teknologi Informasi, perkembangan teknologi salah satunya di bidang *android*. Namun dalam kenyataannya sebagian besar guru masih belum dapat memanfaatkan perkembangan teknologi di bidang *android* ke dalam metode pembelajaran dalam sehari – hari. Berdasarkan hasil pengamatan, anak-anak SD khususnya masih menganggap aksara Jawa sulit dikuasai (Sutarsih, 2015) dan dipelajari pelafalan dan bentuknya, penyebabnya adalah para guru masih mengalami berbagai kesulitan dalam memberikan penjelasan dalam

mengenalkan aksara Jawa kepada siswa-siswinya karena belum adanya media yang mendukung dalam proses belajar mengajar. Bahkan sebagian dari siswa dan siswi sudah beranggapan bahwa belajar aksara Jawa merupakan materi yang sangat sulit dan faktor di lingkungan sekitar menjadi penyebab yang paling utama terhadap keadaan tersebut, media yang paling berperan, media cetak ataupun media elektronik yang sudah jarang sekali menggunakan dan memanfaatkan bahasa Jawa (Avianto and Prasida, 2018).

Media pembelajaran yang berbasis multimedia untuk pengenalan aksara Jawa sebelumnya dilakukan menggunakan pencocokan antar gambar yang ditulis dengan gambar yang sudah disediakan (Arismadhani, 2013) tanpa menggunakan algoritma, menggunakan algoritma *Support Vector Machine (SVM)* (Louis et al., 2019), penelitian dengan algoritma *Learning Vector Quantization (LVQ)* (Agustina, Suwarno and Probeykti, 2011) dan menggunakan algoritma *K-Nearest Neighbor (K-NN)* (Susanto et al., 2018), Untuk meningkatkan kualitas citra dan juga mendapatkan hasil yang terbaik, diusulkan melakukan kolaborasi algoritma K-NN dengan deteksi tepi *Canny*. *Canny* adalah suatu metode deteksi tepi yang cepat dan mudah untuk digunakan (Susanto et al., 2017). Selain itu juga masih banyak lagi penelitian yang mengusulkan untuk pembelajaran secara interaktif tentang pengenalan aksara Jawa untuk kalangan siswa Sekolah Dasar berbasis multimedia ataupun sejenisnya untuk membuat para siswa tertarik dalam mempelajari mengenal aksara Jawa.

Dari penjelasan di atas maka dikembangkan sebuah aplikasi pembelajaran pengenalan aksara Jawa dan akan diimplementasikan untuk pembelajaran mandiri siswa SD di rumah. Serta untuk mengetahui pengaruh tingkat akurasi deteksi tulisan aksara Jawa untuk latihan dengan menggunakan Algoritma K-NN.

METODE PENELITIAN

2.1 Aksara Jawa

Aksara Jawa adalah bagian dari budaya Jawa yang dimiliki masyarakat Jawa. Tulisan aksara Jawa terjadi dengan berbagai sebab, akan tetapi banyak cerita dari sejarah latar belakang penciptaan aksara Jawa tersebut. Isi yang terkandung dalam cerita terdapat banyak

makna yang kental dan berfilosofi tentang ajaran yang sangat luhur, tentang bagaimana dalam mengemban amanah, berjiwa ksatria, menjunjung tinggi kejujuran dan terdapat banyak makna filosofi lainnya yang terdapat pada cerita sejarah tersebut (Arismadhani, 2013).

Kehadiran dari aksara Jawa bukan terjadi dengan sendirinya, aksara Jawa tidak lepas dari tokoh terkenal yang bernama Ajisaka. Huruf-huruf dari aksara Jawa apabila dirangkaikan membentuk urutan kisah atau alur sebuah cerita ringkas dari kesetiaan dua orang pengawal Ajisaka dalam sangat memegang teguh dalam menjalankan perintah dari tuannya Ajisaka hingga keduanya mati.



Gambar 1 : Aksara Jawa

Ha Na Ca Ra Ka
 Ana utusan (ada utusan)
 Da Ta Sa Wa La
 Padha kekerengan (saling berselisih pendapat)
 Pa Dha Ja Ya Nya
 Padha digdayane (sama – sama sakti)
 Ma Ga Ba Tha Nga
 Padha dadi bathange (sama – sama akan menjadi mayat).

Aksara Jawa adalah aksara bersegmental yang didasarkan pada huruf konsonan dengan notasi vocal yang diwajibkan akan tetapi sifatnya sekunder. Beda dengan alphabet dimana vokal memiliki status yang sama dengan huruf konsonannya serta abjad untuk penandaan vokalnya bersifat opsional. Sebagai contoh aksara Ga yang mewakili 2 huruf, yakni G dan A, dan merupakan stu suku kata yang utuh bila dibandingkan dengan kata “gadis”. Dengan demikian, terdapat penyingkatan cacah huruf dalam suatu penulisan kata apabila dibandingkan dengan penulisan huruf alphabet yang biasa.

Aksara Jawa memiliki berbagai bentuk dalam penulisan untuk huruf – hurufnya. Pada penelitian ini dibatasi dengan hanya menggunakan huruf dasar, lebih sering dikenal dengan sebutan aksara *Nglegena*. Aksara *Nglegena* mempunyai 20 bentuk model huruf seperti yang terlihat Gambar 1 di atas.

2.2 K-Nearest Neighbor (K-NN)

K-Nearest Neighbor merupakan metode yang digunakan dalam melakukan klasifikasi suatu objek berdasarkan dari data pembelajaran dimana jaraknya yang paling mendekati dari objek dimaksud. Tujuan KNN adalah klasifikasi objek baru yang berdasarkan atribut yang ada dan data latihan. *Classifier* tidak memakai suatu model dalam mencocokkan dan didasarkan hanya pada memori. Diberikannya titik *query*, kemudian ditemukannya sejumlah k objek atau (titik *training*) terdekat dari titik *query*. Klasifikasi yang digunakan *voting* paling banyak diantara klasifikasi dari k objek. Dalam algoritma K-NN digunakan klasifikasi berketetanggaan sebagai prediksi nilai *query instance* baru. Metode algoritma K-NN sangat sederhana, bekerja dengan berdasarkan jarak paling pendek dari *query instance* ke *training sample* dalam menentukan K-NN-nya(Lestari, 2014).

$$d_{ij} = \sqrt{\sum_{k=1}^n (x_{ik} - x_{jk})^2}$$

Gambar 2 Rumus K-NN

- Keterangan :
- d = Jarak
 - Xi = Data Uji / Testing
 - Xj = Sampel Data
 - N = Dimensi Data
 - k = Variabel Data

Data pembelajaran akan diproyeksikan ke dalam ruang yang berdimensi banyak, masing-masing dari dimensi merupakan representasi karakteristik data tersebut. Ruang kemudian pecah ke bagian-bagian tersendiri berdasarkan hasil klasifikasi pembelajaran data. Jarak tetangga dihitung menurut jarak *Euclidean* dengan menggunakan rumus yang umum.

Karena metode klasifikasi menggunakan metode K-NN maka proses dalam pengaturan

parameter ketetanggaan (k) dan menggunakan rumus jarak sehingga memperoleh hasil akurasi yang maksimal. Jarak yang terdekat dari *basis data* suatu kelas yang didapatkan digunakan untuk patokan penentu data uji termasuk ke dalam jenis kelas yang mana begitu juga dengan hasil yang diperoleh. Klasifikasi K-NN memiliki beberapa langkah, yaitu :

- a Menentukan parameter K (jumlah tetangga terdekat).
- b Hitung kuadrat dari jarak masing–masing objek terhadap *data sample* yang diberikan.
- c Urutkan objek – objek ke dalam kelompok yang memiliki jarak paling kecil.
- d Dengan kategori yang paling banyak, maka dapat diprediksi nilai *instance* yang telah dihitung.
- e Keluaran klasifikasi dari data sampel baru.

Penentuan kelas dilakukan dengan membandingkan jumlah kelas dalam sejumlah k data terdekat dengan data uji. Ukuran untuk kedekatan dari data diuji menggunakan rumus jarak. Rumus jarak yang digunakan sebagai berikut :

a. *Euclidean Distance*

$$d = |x_2 - x_1| + |y_2 - y_1|$$

- d = Jarak
- X₂ = Data Baru
- X₁ = Data Test
- Y₂ = Data Baru2
- Y₁ = Data Test2

b. *Manhattan Distance*

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- d = Jarak
- X₂ = Data Baru
- X₁ = Data Test
- Y₂ = Data Baru2
- Y₁ = Data Test2

K-NN mengklasifikasi dengan proyeksi pembelajaran data pada ruang yang berdimensi banyak. Ruang ini kemudian dibagi ke dalam bagian–bagian yang menggambarkan kriteria data pembelajaran. Setiap dari data

pembelajaran diwujudkan menjadi titik–titik c pada ruang berdimensi banyak.

Data baru yang sudah diklasifikasi kemudian diproyeksikan di ruang berdimensi banyak yang terdapat titik–titik k data pembelajaran. Melakukan proses klasifikasi dengan mencari titik k terdekat dari titik k-baru (*Nearest Neighbor*). Teknik mencari tetangga yang terdekat umumnya menggunakan rumus jarak *Euclidean*.

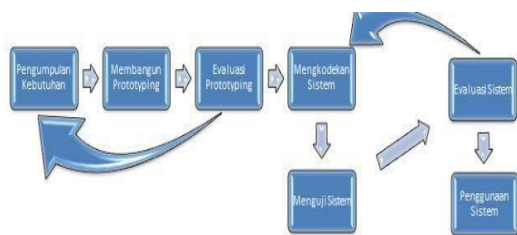
Dalam mengimplementasikan algoritma *K-Nearest Neighbor*, banyaknya k tetangga yang terdekat digunakan untuk mengklasifikasi data yang baru perlu ditentukan terlebih dahulu. Jumlah k, sebaiknya disarankan menggunakan angka ganjil. Dipertimbangkan dalam menentukan nilai k berdasarkan jumlah data yang sudah ada dan memperhatikan ukuran dari dimensi yang terbentuk oleh data. Semakin banyak jumlah data yang dimiliki, angka k yang akan dipilih disarankan semakin lebih rendah. Namun jika dimensi data semakin besar ukurannya, angka k yang dipilih sebaiknya yang semakin tinggi.

2.3 Metode Pengembangan Sistem

Pada penelitian ini Metode pengembangan sistem yang digunakan adalah metode *Prototyping*(Sommerville, 2011). Dalam metode *Prototyping* terdapat sejumlah aktivitas yaitu:

- a. Pengumpulan Kebutuhan
- b. Membangun Prototype
- c. Evaluasi Prototype
- d. Mengkodekan Sistem
- e. Menguji Sistem
- f. Evaluasi Sistem
- g. Menggunakan Sistem

Adapun gambaran metode *Prototyping* dapat digambarkan urutannya seperti tampak pada Gambar 3 di bawah ini



Gambar 3 Prototyping Model

HASIL DAN PEMBAHASAN

3.1. Implementasi K-NN

Dalam aplikasi mengklasifikasi Aksara Jawa dengan algoritma *K-Nearest Neighbor* berbasis android mengambil rumus jarak yang tingkat validnya maksimum yaitu *Euclidean Distance*.

Dalam perhitungan ini terdapat 3 tahap dalam implementasi metode dalam aplikasi yaitu

a. Perhitungan Jarak

Berikut menjelaskan tahap 1 tentang perhitungan jarak dengan gambar 4.12. Method ini digunakan untuk menghitung jarak dengan menggunakan rumus jarak *Euclidean Distance*. Gambar 4 sebagai berikut :

```

private fun countDistance(listData: ArrayList<DataModel?>, newDataModel: DataModel?): ArrayList<DataModel?> {
    for (i in listData?.indices!!) {
        Log.d(tag, map: "listData" + listData[i])

        //Pixel from new data
        val newRedPixel = newDataModel?.rgb?.redPixel
        val newGreenPixel = newDataModel?.rgb?.greenPixel
        val newBluePixel = newDataModel?.rgb?.bluePixel

        //Pixel from data training
        val redPixel = listData[i]?.rgb?.redPixel
        val greenPixel = listData[i]?.rgb?.greenPixel
        val bluePixel = listData[i]?.rgb?.bluePixel

        Log.d(tag, map: "Pixel $newRedPixel $redPixel")

        //difference of pixel (k-n)^2
        val differenceRed = (newRedPixel!! - redPixel!!).pow(2.toDouble())
        val differenceGreen = (newGreenPixel!! - greenPixel!!).pow(2.toDouble())
        val differenceBlue = (newBluePixel!! - bluePixel!!).pow(2.toDouble())

        var distance = sqrt((differenceRed + differenceGreen + differenceBlue)
        distance = HelperClass().setNumberBehindComa(distance)

        listData[i]?.classification?.distance = distance

        Log.d(tag, map: "Distance ${listData[i]?.classification?.id} ${listData[i]?.classification?.distance}")
    }
    return listData
}
    
```

Gambar 4. Perhitungan Jarak

b. Pengurutan Jarak Terpendek

Gambar 5 menjelaskan tahap 2 mengenai pengurutan jarak terpendek dengan mengurutkan mulai dari jarak terdekat ke terjauh. *listData* merupakan daftar yang akan diurutkan.

```

private fun sortList(listData: ArrayList<DataModel?>): List<DataModel?> {
    return listData?.sortedBy { it?.classification?.distance }
}
    
```

Gambar 5. Pengurutan Jarak Terpendek

c. Menentukan Hasil Klasifikasi

Tahap ketiga ini merupakan tahap terakhir dalam implementasi metode, yaitu menentukan hasil dari data baru apakah bernilai *true* atau *false* dengan cara membandingkan jumlah data yang bernilai *true* dan *false* sebanyak k. K adalah angka penentu untuk mendapatkan hasil yang akan dilakukan

perbandingan. Gambar 6 potongan *code* yang digunakan untuk menentukan hasil klasifikasi.

```
private fun specifyResult(listData: List<DataModel?>, k: Int, aksara: String?): String {
    val listAksara = ListHanacaraka().getListAksaraCount()
    for (i in 0 until k) {
        when(listData[[i]].classification?.aksara){
            "HA" -> listAksara[0].total = listAksara[0].total+1
            "NA" -> listAksara[1].total = listAksara[1].total+1
            "CA" -> listAksara[2].total = listAksara[2].total+1
            "RA" -> listAksara[3].total = listAksara[3].total+1
            "KA" -> listAksara[4].total = listAksara[4].total+1
            "DA" -> listAksara[5].total = listAksara[5].total+1
            "TA" -> listAksara[6].total = listAksara[6].total+1
            "SA" -> listAksara[7].total = listAksara[7].total+1
            "WA" -> listAksara[8].total = listAksara[8].total+1
            "LA" -> listAksara[9].total = listAksara[9].total+1
            "PA" -> listAksara[10].total = listAksara[10].total+1
            "DHA" -> listAksara[11].total = listAksara[11].total+1
            "JA" -> listAksara[12].total = listAksara[12].total+1
            "YA" -> listAksara[13].total = listAksara[13].total+1
            "MA" -> listAksara[14].total = listAksara[14].total+1
            "GA" -> listAksara[15].total = listAksara[15].total+1
            "BA" -> listAksara[16].total = listAksara[16].total+1
            "THA" -> listAksara[17].total = listAksara[17].total+1
            "NGA" -> listAksara[18].total = listAksara[18].total+1
        }
    }
    val jumlahAksara = listAksara?.filter { it.aksara==aksara }?[0].total
    listAksara?.sortByDescending { it.total }
    return if (listAksara[0].total==jumlahAksara) aksara.toString()else listAksara[0].aksara.toString()
}
```

Gambar 6. Menentukan Hasil Klasifikasi

3.2. Implementasi Aplikasi

Aplikasi berbasis android pengklasifikasian Aksara Jawa berbasis Android yang berfungsi untuk media pembelajaran siswa Sekolah Dasar, agar dalam mempelajari pengenalan Aksara Jawa lebih mudah dan menarik untuk mempelajarinya terutama ketika belajar mandiri di rumah.

Aplikasi pertama kali dijalankan ada menampilkan halaman pembuka dan diikuti oleh menu pilihan antara lain adalah menu Sinau Nulis, menu Kuis, menu Daftar Skor dan menu Tentang Aplikasi, dapat dilihat di Gambar 6 tampilan halaman awal aplikasi.



Gambar 7. Halaman Depan

Selanjutnya jika menginginkan melihat daftar aksara Jawa dapat memilih menu daftar aksara, jika ingin belajar menulis pilih menu belajar menulis, pada Gambar 8 adalah tampilan dari daftar Aksara Jawa dan formulir untuk Latihan menulis.



Gambar 8. Daftar Aksara Jawa & Belajar Menulis

Kemudian untuk melakukan pengujian siswa dapat memilih menu kuis, diberikan 10 soal yang dapat dikerjakan, setelah selesai akan ditampilkan hasil skor yang diperoleh dan akan tersimpan sehingga akan dapat dilihat perkembangan anak dalam belajar aksara Jawa, ditampilkan yang mendapat nilai tertinggi terlihat pada Gambar 9 berikut :



Gambar 9. Kuis dan Nilai

3.3. Pengujian

Hasil pengujian K-NN berdasarkan parameter yang berbeda, yaitu k = 3, k = 5, k = 7 mendapatkan hasil sebagai berikut :

Tabel 1. Pengujian K-NN

Hasil Pengujian	Hasil
-----------------	-------

	Penguji n		
	K = 3	K = 5	K = 7
Klasifikasi Benar	528	566	533
Klasifikasi Salah	494	458	491
% Berhasil	51,5 %	54,9 %	52,1 %
% Gagal	48,5 %	44,1 %	47,9 %

Tabel 2 berikut ini menjelaskan tentang bagaimana prosentase kebenaran yang dilakukan terhadap sampel siswa, siswa yang dijadikan sampel adalah 10 anak yaitu sebagai berikut :

Tabel 2. Prosentasi Hasil Belajar

Nama	Benar	Salah
Gymnas Tiar	8	2
Bagas Aji Saputra	7	3
Sri Wahyuningsih	8	2
Alfi Mar'atus	8	2
Niya Ani Kusuma	9	1
Deni Sudargo	5	5
Silvia Ayu Perrmata	10	0
Cindy Amalia P	7	7
Dika Prasetyo	10	0
Ariska Dewi	8	2
Alfiatun Khasanah	6	4
Prosentase (%)	86 %	14 %

KESIMPULAN

Aplikasi mengklasifikasi Aksara Jawa dengan algoritma *K-Nearest Neighbor* berbasis android sudah dapat digunakan oleh pengguna setelah dilakukan pengujian berdasarkan nilai k yang paling akurat adalah k = 5 dengan hasil berhasil 54,9% dan hasil gagal 45,1% serta pengujian kebenaran siswa dengan hasil benar 86% dan hasil salah 14%. Dengan hasil tersebut aplikasi dapat digunakan oleh pengguna khususnya untuk anak SD untuk mempelajari Aksara Jawa dan belajar menulis Aksara Jawa dengan mudah.

DAFTAR PUSTAKA

Agustina, A., Suwarno, S. and Proboyekti, U. (2011) 'Pengenalan Aksara Jawamenggunakan Learning Vector Quantization (Lvq)', *Informatika: Jurnal Teknologi Komputer dan Informatika*, (April).

Arismadhani, A. et al. (2013) 'Aplikasi Belajar Menulis Aksara Jawa Menggunakan Android', *Jurnal Teknik Pomits*, 2(1), p. 1. Available at: <http://ejurnal2.its.ac.id/index.php/teknik/article/view/2732>.

Avianto, Y. F. and Prasida, T. A. S. (2018) 'Pembelajaran Aksara Jawa Untuk Siswa Sekolah Dasar Dengan Menggunakan Media Board Game', *Aksara*, 30(1), p. 133. doi: 10.29255/aksara.v30i1.223.133-148.

GlobalStats (2020) *Mobile Operating System Market Share Indonesia*. Available at: <https://gs.statcounter.com/os-market-share/mobile/indonesia> (Accessed: 14 April 2020).

Huri, D. (2014) 'Penguasaan Kosakata Kedwibahasaan Antara Bahasa Sunda dan Bahasa Indonesia Pada Anak-Anak', *Jurnal Pendidikan Unsika*, 2(November), pp. 88–93.

Kemp, S. (2020) 'Digital 2020 :Global Digital Overview', *Global Digital Insights*, p. 247. doi: <https://datareportal.com/reports/digital-2020-global-digital-overview>.

Lestari, M. (2014) 'Penerapan Algoritma Klasifikasi Nearest Neighbor (K-NN) untuk Mendeteksi Penyakit Jantung', *Faktor Exacta*, 7(September 2010), pp. 366–371.

Louis, A. et al. (2019) 'Pengenalan Aksara Jawa dengan Menggunakan Metode Area Based Feature Extraction dan Support Vector Machine', *Jurnal Infra Petra*.

Sommerville, I. (2011) *Software Engineering (Rekayasa Perangkat Lunak)*. Jakarta: Erlangga.

Susanto, A. et al. (2017) 'Perlindungan Hak Cipta Pada Citra Digital Menggunakan', *Simetris : Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, 8(2), pp. 441–448.

Susanto, A. et al. (2018) 'A High Performace of Local Binary Pattern on Classify Javanese Character Classification', *Scientific Journal of Informatics*, 5(1), p. 8. doi: 10.15294/sji.v5i1.14017.

Sutarsih (2015) 'Pembelajaran Menulis Aksara Jawa Anak Kelas III Sekolah Dasar.', *ISSN 0854-3283 Vol. 27, No. 1, Juni 2015*, 65., 27(1), pp. 65–72.