

# Pemilihan Parameter Smoothing pada Probabilistic Neural Network dengan Menggunakan Particle Swarm Optimization untuk Pendeteksian Teks Pada Citra

Endah Ekasanti Saputri

*Fakultas Ilmu Komputer, Universitas Dian Nuswantoro*

*Email: endah.ekasanti@gmail.com*

Romi Satria Wahono dan Vincent Suhartono

*Fakultas Ilmu Komputer, Universitas Dian Nuswantoro*

*Email: romi@brainmatics.com, vincent.suhartono@dsn.dinus.ac.id*

*Abstract:* Teks sering dijumpai di berbagai tempat seperti nama jalan, nama toko, spanduk, penunjuk jalan, peringatan, dan lain sebagainya. Deteksi teks terbagi menjadi tiga pendekatan yaitu pendekatan tekstur, pendekatan edge, dan pendekatan Connected Component. Pendekatan tekstur dapat mendeteksi teks dengan baik, namun membutuhkan data training yang banyak. Probabilistic Neural Network (PNN) dapat mengatasi permasalahan tersebut. Namun PNN memiliki permasalahan dalam menentukan nilai parameter smoothing yang biasanya dilakukan secara trial and error. Particle Swarm Optimization (PSO) merupakan algoritma optimasi yang dapat menangani permasalahan pada PNN. Pada penelitian ini, PNN digunakan pada pendekatan tekstur guna menangani permasalahan pada pendekatan tekstur, yaitu banyaknya data training yang dibutuhkan. Selain itu, digunakan PSO untuk menentukan parameter smoothing pada PNN agar akurasi yang dihasilkan PNN-PSO lebih baik dari PNN tradisional. Hasil eksperimen menunjukkan PNN dapat mendeteksi teks dengan akurasi 75,42% hanya dengan menggunakan 300 data training, dan menghasilkan 77,75% dengan menggunakan 1500 data training. Sedangkan PNN-PSO dapat menghasilkan akurasi 76,91% dengan menggunakan 300 data training dan 77,89% dengan menggunakan 1500 data training. Maka dapat disimpulkan bahwa PNN dapat mendeteksi teks dengan baik walaupun data training yang digunakan sedikit dan dapat mengatasi permasalahan pada pendekatan tekstur. Sedangkan, PSO dapat menentukan nilai parameter smoothing pada PNN dan menghasilkan akurasi yang lebih baik dari PNN tradisional, yaitu dengan peningkatan akurasi sekitar 0,1% hingga 1,5%. Selain itu, penggunaan PSO pada PNN dapat digunakan dalam menentukan nilai parameter smoothing secara otomatis pada dataset yang berbeda.

*Keywords:* deteksi teks, pendekatan tekstur, probabilistic neural network, particle swarm optimization, parameter smoothing

## 1 PENDAHULUAN

Teks atau tulisan sering kita jumpai di berbagai tempat seperti nama jalan, nama toko, spanduk, penunjuk jalan, peringatan, dan lain sebagainya. Teks memiliki peranan penting dalam kehidupan sehari-hari karena teks kaya akan informasi (Karaoglu, Fernando, & Tremeau, 2010) (Epshtein, Ofek, & wexler, 2010) (Meng & Song, 2012) (Angadi & Kodabagi, 2010). Hal ini dapat menjadi masalah pada seorang turis yang tidak mengerti bahasa sekitar, serta seorang yang memiliki masalah penglihatan. Untuk itu, saat ini banyak

dibangun suatu perangkat seperti tourist assistance, atau sebuah sistem yang dapat membantu seorang dengan masalah penglihatan untuk dapat mengelilingi kota dan melakukan aktifitas sehari-hari. Situasi perkotaan dapat dianalisa secara simultan dan ditambah dengan algoritma text-to-speech, membuat perangkat tersebut dapat membaca tanda-tanda jalan, label pada pusat perbelanjaan, dan sebagainya (Karaoglu, Fernando, & Tremeau, 2010). Sehingga perangkat tersebut diharapkan dapat membantu seorang yang memiliki masalah penglihatan.

Ketersediaan perangkat yang murah seperti handphone, menjadikan deteksi teks pada citra menjadi luas. Karena pengambilan citra dengan kamera memiliki banyak masalah seperti orientasi, pencahayaan, low resolution, blur, menjadikan deteksi teks menjadi penelitian yang menarik. Dan mendorong para peneliti untuk mencoba mencari solusi untuk menemukan desain sistem sesederhana mungkin yang dapat menangani permasalahan yang ada (Karaoglu, Fernando, & Tremeau, 2010). Selain itu, menurut penelitian (Karaoglu, Fernando, & Tremeau, 2010) (Meng & Song, 2012) (Shivakumara, Phan, & Tan, 2011), deteksi teks pada citra juga memiliki banyak permasalahan seperti ukuran tulisan, jenis tulisan dan warna tulisan yang berbeda, serta background yang kompleks.

Berdasarkan penelitian (Shivakumara, Phan, & Tan, 2011) (Shi, Xiao, Wang, & Zhang, 2012) (Angadi & Kodabagi, 2010), metode deteksi teks pada citra alam, secara umum dapat dikategorikan menjadi tiga, yaitu pendekatan Connected Component (CC), batas tepi (edge based approach), dan tekstur. (Karaoglu, Fernando, & Tremeau, 2010) melakukan penelitian dengan pendekatan CC, namun CC tidak dapat bekerja dengan baik pada garis teks yang memiliki kontras yang rendah (garis teks yang terputus-putus) serta bermasalah pada complex background. Permasalahan kontras yang rendah pada CC ini, dapat ditangani oleh pendekatan batas tepi. (Bai, Yin, & Liu, 2012) melakukan penelitian dengan menggunakan *edge strength* dan *variance of orientation* namun pendekatan ini juga memiliki permasalahan pada tingginya nilai komputasi pada dataset yang besar. Dilain pihak, (Huang, 2012) melakukan penelitian dengan menggunakan pendekatan tekstur yang dapat menangani permasalahan pada pendekatan batas tepi dan *complex background* pada CC dengan cara mempertimbangkan bahwa teks memiliki tekstur yang berbeda. Namun pendekatan tekstur membutuhkan jumlah sample yang besar untuk data training teks dan non-teks.

Pada penelitian ini, akan digunakan pendekatan tekstur dalam mendeteksi teks pada citra alam. Secara umum, ketiga pendekatan tersebut memiliki beberapa tahapan, yaitu

*candidate text region extraction, text line localization, feature extraction, training*, verifikasi (Karaoglu, Fernando, & Tremeau, 2010). *Feature extraction* yang biasa digunakan oleh para peneliti khususnya pada pendekatan tekstur, adalah Gabor Wavelet. Seperti penelitian yang dilakukan oleh (Shivakumara & Tan, 2010) (Wang & Wang, 2010) (Angadi & Kodabagi, 2010) (Le, Dinh, Kim, & Lee, 2010) (Huang, 2012) (Lee, Lee, Lee, Yuille, & Koch, 2011). Sedangkan pada tahap verifikasi, dibutuhkan suatu algoritma / metode *learning* untuk membedakan citra teks dan non-teks dari citra yang dihasilkan pada tahap sebelumnya.

Algoritma yang biasanya digunakan pada pendekatan texture dalam mendeteksi teks adalah Support Vector Machine(SVM), Random Forest, K-Means, dan ANN. SVM dapat menghasilkan akurasi yang tinggi, dan dapat menangani kasus dengan dimensi besar, namun membutuhkan data training yang besar. Sedangkan Random Forest, memiliki kelebihan pada proses *learning* yang cepat, dapat menangani input variabel yang besar, dan dapat menghasilkan akurasi tinggi. Namun Random Forest ini bermasalah pada waktu komputasi yang lama. Selain itu, K-Means memiliki kelebihan dalam menangani background yang kompleks. Namun bermasalah pada kontras yang rendah dan waktu komputasi yang tinggi.

Probabilistic Neural Network (PNN) merupakan algoritma klasifikasi dan merupakan suatu algoritma ANN yang menggunakan fungsi *probabilistic*, tidak membutuhkan dataset yang besar dalam tahap pembelajarannya, serta memiliki kelebihan yaitu dapat mengatasi permasalahan yang ada pada Back-Propagation(BP) yaitu dapat mengatasi waktu pelatihan yang lama, terjebak pada global minimum, dan sulitnya perancangan arsitektur jaringan (Spech, 1990). Berdasarkan paper yang ada, PNN dapat digunakan untuk mengklasifikasikan secara akurat pada beberapa penelitian serta memiliki beberapa kelebihan dibandingkan dengan BP. Sehingga, algoritma ini diharapkan dapat digunakan untuk memperbaiki kekurangan yang ada pada pendekatan tekstur, yaitu dengan menggunakan PNN pada tahap verifikasi citra. Namun, PNN memiliki masalah pada penentuan parameter smoothing yang biasanya ditentukan dengan cara *trial and error* atau *user\_defined* (Yang & Yang, 2012). Untuk itu, diperlukan suatu metode optimasi yang dapat menentukan parameter *smoothing* yang paling optimal pada PNN.

Pada beberapa penelitian, penentuan parameter *smoothing* pada PNN dioptimasi dengan menggunakan Genetic Algorithm(GA) (Yang & Yang, 2012), berdasarkan Centre Neighbor (Liu, Wang, & Cheng, 2011) menggunakan PSO dalam menentukan parameter smoothing pada PNN dan hasil eksperimen menunjukkan tingkat akurasi yang dihasilkan oleh PSO-PNN lebih tinggi dari tradisional PNN, yaitu 63,31% untuk tradisional PNN dan 93,95% untuk PSO-PNN. Sehingga PSO diharapkan dapat mengatasi masalah yang ada pada PNN, yaitu dalam menentukan nilai *smoothing* yang sesuai, sehingga dapat menghasilkan klasifikasi teks yang lebih akurat.

Pada penelitian ini, akan menggunakan pendekatan tekstur, yaitu dengan menerapkan Probabilistic Neural Network(PNN) dan menerapkan Particle Swarm Optimization(PSO) untuk menentukan parameter *smoothing* yang ada pada PNN. Sehingga diharapkan dapat menghasilkan akurasi yang lebih akurat.

## 2 PENELITIAN TERKAIT

Permasalahan yang ada pada deteksi teks, yaitu banyaknya variasi warna, orientasi, pencahayaan, *low resolution, blur, noise*. Selain itu, banyak metode yang diusulkan yang belum

dapat menangani permasalahan tersebut dengan baik. Seperti penelitian yang dilakukan (Angadi & Kodabagi, 2010). Latar belakang dari penelitian yang dilakukan Angadi adalah kurang efisiennya pendekatan Connected Component (CC) dan pendekatan edge dalam mengatasi gambar yang memiliki *noise, low resolution image* dan *complex background*. Penelitian yang dilakukan Angadi, menggunakan Discrete Cosine Transform (DCT) pada tahap *preprocessing*, Homogeneity Function pada tahap *Feature Extraction*, dan pada tahap *classification* menggunakan Discriminant Function, dan menghasilkan akurasi sebesar 96,6%.

Selain itu, (Ji, Xu, Yao, Zhang, Sun, & Liu, 2008) melakukan penelitian berdasarkan permasalahan pada pencahayaan dan kontras yang tidak konsisten pada gambar yang akan dideteksi. Penelitian ini menggunakan Pyramid Haar pada tahap *preprocessing*, Haar Wavelet pada tahap *Feature Extraction*, dan selanjutnya menggunakan Directional Correlation Analysis (DCA) pada tahap *classification*. Penelitian ini menghasilkan precision (p) 0.50, recall (r) 0.79, dan system performance (f) 0.68.

Sedangkan penelitian yang dilakukan oleh (Pan, Liu, & Hou, 2010) dilandasi oleh hasil kandidat teks pada pendekatan CC yang masih terlalu kasar dan dapat mempersulit proses *verification*. Penelitian yang dilakukan Pan, menggunakan boosted classifier dan polynomial classifier pada tahap *preprocessing*, Gabor Wavelet pada tahap Feature Extraction, dan Histogram of Oriented Gradients, Local Binary Pattern, Discrete Cosine Transform pada tahap *classification*.

Pada penelitian ini menggunakan pendekatan texture yang dapat menangani permasalahan pada deteksi teks yaitu pada *low resolution image*, blur, pencahayaan, kontras, variasi warna, ukuran dan jenis tulisan dengan menggunakan gabungan Probabilistic Neural Network (PNN) dan particle Swarm Optimization (PSO) pada tahap *classification*.

## 3 METODE YANG DIUSULKAN

Pada penelitian ini, data yang digunakan adalah dataset ICDAR Competition 2003 dan ICDAR Competition 2011. Sedangkan, metode yang diusulkan pada penelitian ini adalah menggunakan pendekatan tekstur. Gambar alam yang akan dideteksi, dipecah atau dipotong-potong menjadi blok-blok kecil dengan ukuran 70x70 pixel dan 100x100 pixel. Kemudian, blok tersebut diubah kedalam *grayscale image* yang selanjutnya diambil cirinya dengan metode *feature extraction* Gabor Wavelet. Setelah itu, ciri dari blok-blok tersebut dikenali sebagai teks atau non teks dengan metode *classification* Probabilistic Neural Network (PNN) yang telah di-*improved* dengan suatu metode optimasi Particle Swarm Optimization (PSO). Masalah pada penelitian ini adalah sulitnya menentukan nilai parameter *smoothing* pada PNN. Sedangkan nilai parameter *smoothing* pada PNN ini sangat menentukan tingginya tingkat akurasi yang dihasilkan oleh PNN dalam mendeteksi teks.

Tahapan pada proses *training* PNN adalah:

1. Menghitung Total Minimum Distance (TMD) pada masing-masing kelas dengan menggunakan Persamaan:

$$TMD = \sum_{i=1}^n |x_1(i) - x_2(i)|$$

2. Kemudian menghitung Parameter Smoothing pada setiap kelas dengan rentang nilai  $g$  sesuai inputan *user* menggunakan Persamaan:

$$S = \frac{g * TMD_0}{\text{jumlah pola latih kelas} - n}$$

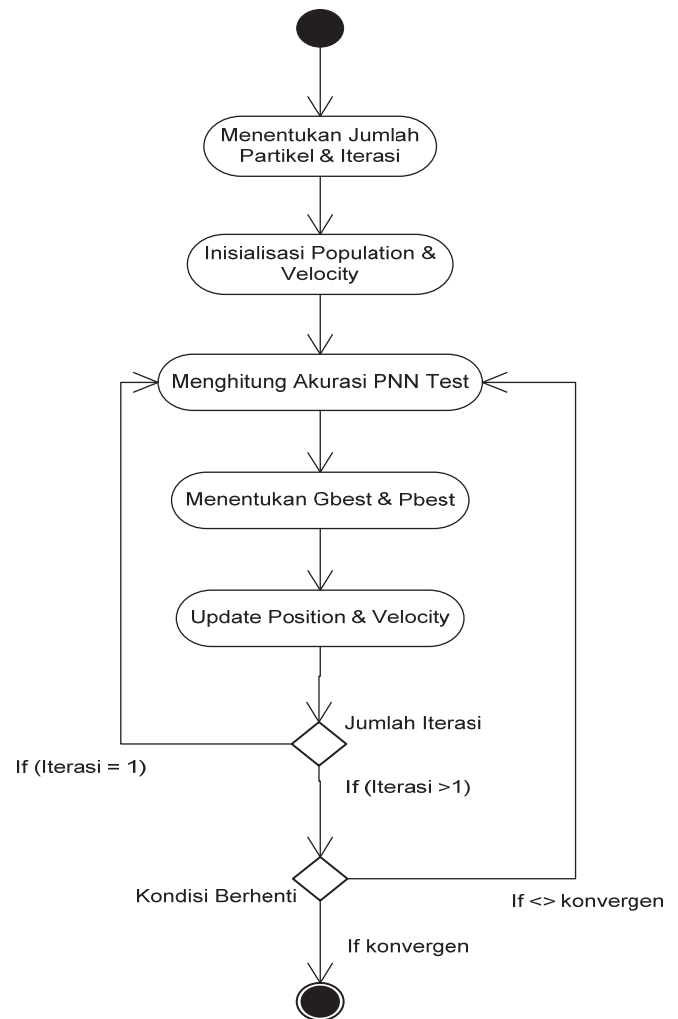
3. Setelah itu, menghitung probabilitas setiap pola terhadap kedua kelas dengan Persamaan :

$$p(x|C_k) = \frac{1}{(2\pi)^{m/2} \sigma_k^m |C_k|} \sum_{\rho_i \in C_k} \exp[-\|x - w_i\|^2 / (2\sigma_k^2)]$$

4. Dan yang terakhir, mencari nilai maksimum dari probabilitas kedua kelas pada suatu pola.

Sedangkan pada proses *testing*, PNN menghitung Total Minimum Distance (TMD) antara pola *testing* dengan pola *training* sesuai dengan kelasnya. Dengan menggunakan parameter *smoothing* yang sudah dihitung pada tahap *training*, kemudian PNN menghitung probabilitas pola terhadap kedua. Berdasarkan probabilitas tersebut, dicari nilai maksimum dari probabilitas kedua kelas.

Pemilihan parameter *smoothing* pada Probabilistic Neural Network (PNN) biasanya dilakukan secara *trial and error*. Seperti pada Gambar 1, penelitian ini menggunakan Particle Swarm Optimization (PSO) dalam menentukan nilai dari parameter *smoothing*. Nilai parameter *smoothing* yang digunakan diperoleh dari perhitungan PSO terhadap akurasi pada PNN Test. Nilai parameter *smoothing* yang menghasilkan akurasi PNN Test terbaiklah yang digunakan pada proses selanjutnya.



Gambar 1. Diagram Aktifitas PNN-PSO

Akurasi yang dihasilkan dihitung menggunakan *confusion matrix*. Perhitungan pada *confusion matrix* dihitung berdasarkan prediksi *positif* yang benar (*True Positif*), prediksi *positif* yang salah (*False Positif*), prediksi negatif yang benar (*True Negatif*) dan prediksi *negatif* yang salah (*False Negatif*).

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN}$$

Semakin tinggi nilai akurasinya, semakin baik pula metode yang dihasilkan.

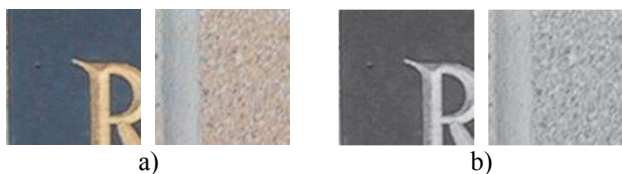
#### 4 HASIL EKSPERIMEN

Eksperimen dilakukan dengan menggunakan Matlab R2009b pada Intel Core 2 Duo, 1GB RAM, 80GB HD, dan system operasi Windows XP SP1.

Pada penelitian ini, gambar yang menjadi inputan awal adalah gambar alam yang memiliki teks atau tulisan di dalamnya, seperti Gambar 2. Kemudian gambar tersebut masuk pada tahap *preprocessing* yaitu pemotongan gambar menjadi 100x100 pixel (Gambar 3a) dan mengubah gambar yang sudah dipotong tersebut menjadi *grayscale image* (Gambar 3b) untuk memperkecil dimensi warna yang terkandung di dalam gambar tersebut.



Gambar 2. Gambar yang mengandung teks



Gambar 3. Blok gambar 100x100 pixel

Setelah tahap *preprocessing*, gambar tersebut diambil cirinya dengan metode ekstraksi ciri Gabor Wavelet, yaitu dengan membangun matrix konvolusi, yaitu *real* dan *imaginer*, dan dioperasikan dengan matrix asli. Sehingga menghasilkan matrix ciri yang digunakan pada tahap selanjutnya, yaitu *classification*.

Proses *training* pada PNN menghasilkan nilai *Total Minimum Distance*(TMD) yang digunakan pada proses selanjutnya, yaitu tahap validasi dan *testing*. Seperti pada Tabel 2 berikut, dengan mengubah-ubah jumlah data training yang digunakan, akan berbeda pula nilai TMD yang dihasilkan.

Tabel 2. Total Minimum Distance Hasil Proses Training

Jumlah Data Training	300 (Positif = 100; Negatif = 200)	600 (Positif = 200; Negatif = 400)	900 (Positif = 300; Negatif = 600)	1200 (Positif = 400; Negatif = 800)	1500 (Positif = 500; Negatif = 1000)
TMD kelas 1 (teks)	23.50	40.31	59.25	77.58	97.48
TMD kelas 2 (non-teks)	13.93	23.61	34.36	41.91	51.49

Eksperimen dilakukan dengan melakukan percobaan pada jumlah data yang digunakan pada tahap *training* PNN. Percobaan ini dilakukan untuk mengetahui seberapa berpengaruh jumlah data yang digunakan pada tahap training dan kombinasinya terhadap parameter *smoothing* dengan akurasi yang dihasilkan system.

Setelah dilakukan eksperimen terhadap PNN, nilai parameter *smoothing* yang sesuai pada studikusus dan data *training* yang digunakan adalah antara 0,8 dan 1. Sesuai penelitian yang dilakukan (Liu, Wang, & Cheng, 2011) mengatakan bahwa nilai parameter *smoothing* dapat berubah ubah sesuai dengan data *training* yang digunakan.

Selanjutnya, kita bandingkan akurasi yang dihasilkan. Grafik pada Gambar 7 menunjukkan akurasi yang dihasilkan PNN-PSO lebih baik dari PNN tradisional. Jika dihitung secara rata-rata, akurasi yang dihasilkan PNN adalah 74,7%, sedangkan PNN-PSO menghasilkan akurasi 77,12%.

Tabel 4. Hasil Komparasi PNN dan PNN-PSO

Algoritma	Akurasi
PNN	74,7%
PNN-PSO	77,12%



Gambar 7. Grafik Komparasi PNN dan PNN-PSO

Pemilihan jumlah iterasi dan populasi pada PSO juga mempengaruhi akurasi dan waktu yang dibutuhkan dalam mendeteksi teks. Seperti yang diungkapkan (Jorhedi & Jasni, 2013) (Kennedy & Eberhart, 1995) dalam penelitiannya, bahwa jumlah partikel yang terlalu banyak akan menyebabkan algoritma terjebak pada local optima dan dapat memperberat proses komputasi algoritma yang mengakibatkan lamanya waktu komputasi. Jumlah partikel yang menghasilkan akurasi tertinggi dihasilkan dengan jumlah partikel 100.

Sedangkan, *stopping criteria* yang digunakan pada penelitian ini adalah jumlah iterasi. Jumlah iterasi ini disesuaikan dengan kasus yang ditangani pada suatu penelitian. akurasi tertinggi lebih banyak dihasilkan pada jumlah iterasi sama dengan 1000. Seperti pada penelitian (Jorhedi & Jasni, 2013) (Kennedy & Eberhart, 1995) mengatakan bahwa PSO akan membutuhkan iterasi yang lebih banyak untuk mendapatkan global optimum dan menghindari kegagalan dalam menemukan global optimum. Sebab, semakin banyak iterasi yang digunakan pada suatu kasus, akan membuat pergerakan partikel lebih tersebar keseluruhan daerah pencarian.

## 5 KESIMPULAN

Pada penelitian ini dilakukan pengujian model dengan menggunakan Probabilistic Neural Network (PNN) dan Probabilistic Neural Network & Particle Swarm Optimization (PNN-PSO). Beberapa percobaan dilakukan dengan mengkombinasikan beberapa parameter pada PNN dan PSO untuk mendapatkan akurasi terbaik dalam mendeteksi teks pada gambar alam. Parameter PNN yang diujicoba adalah jumlah data *training*, sedangkan parameter PSO yang diujicoba adalah jumlah partikel pada suatu populasi, dan jumlah maksimum iterasi. Hasil percobaan menunjukkan: 1) dalam mendeteksi teks, PNN dapat menghasilkan akurasi 77,75% dengan menggunakan nilai parameter *smoothing* 0,8 dan jumlah data training 1500. 2) Sedangkan PNN-PSO dapat menghasilkan akurasi 77,89% pada jumlah data training 1500, jumlah populasi 50, dan maksimum iterasi 1000. Dari hasil pengujian diatas, dapat disimpulkan bahwa: 1) penggunaan PNN pada pendekatan tekstur dalam mendeteksi teks pada gambar alam tidak harus menggunakan data training yang banyak. 2) Penggunaan PSO dalam menentukan parameter *smoothing* pada PNN dapat meningkatkan akurasi, dan dapat digunakan dalam menentukan parameter *smoothing* secara otomatis pada dataset yang berbeda.

## REFERENSI

- Angadi, S. A. (2010). Text Region Extraction from Low Resolution Natural Scene Images using Texture Features. *International Advance Computing Conference*, 121-128.
- Bai, B., Yin, F., & Liu, C. (2012). A Fast Stroke-Based Method for Text Detection in Video. *IAPR International Workshop on Document Analysis Systems*, 69-73.
- Donald, F. S. (1990). Probabilistic Neural Networks. *Neural Network I Pergamon Press pie Original Contribution*, 3.
- Epshtein, B., Ofek, E., & Wexler, Y. (2010). Detecting text in natural scenes with stroke width transform. *Computer Vision and Pattern Recognition (CVPR)*, 10, 2963-2970.
- Huang, X. (2012). Automatic Video Text Detection and Localization Based on Coarseness Texture. *International Conference on Intelligent Computation Technology and Automation*, 1(2).
- Jamil, A., Siddiqi, I., Arif, F., & Raza, A. (2011). Edge-based Features for Localization of Artificial Urdu Text in Video Images. *International Conference on Document Analysis and Recognition*, 1120-1124.
- Ji, R., Xu, P., Yao, H., Zhang, Z., Sun, X., & Liu, T. (2008). Directional correlation analysis of local Haar binary pattern for text detection. *IEEE International Conference on Multimedia and Expo*, 885-888.
- Jorhedi, R. A., & Jasni, J. (2013). Parameter Selection in Particle Swarm Optimization: a survey. *Journal of Experimental & Theoretical Artificial Intelligence*, 25, 527-542.
- Karaoglu, S., Fernando, B., Trémeau, A., & Etienne, S. (2010). A Novel Algorithm for Text Detection and Localization in Natural Scene Images. *IEEE Digital Image Computing : Techniques and Applications*, 641-648.
- Kennedy, J., & Eberhart, R. (1995). Particle Swarm Optimization. 1942-1948.
- Le, H. P., Dinh, T. N., Kim, S. H., & Lee, G. S. (2010). Text Detection in Binarized Text Images of Korean Signboard by Stroke Width Feature. *IEEE International Conference on Computer and Information Technology (CIT 2010)*, 1588-1592.
- Lee, J., Lee, P., Lee, S., Yuille, A., & Koch, C. (2011). AdaBoost for Text Detection in Natural Scene. *International Conference on Document Analysis and Recognition*.
- Liu, B., Wang, H., & Cheng, X. (2011). Exchange Rate Forecasting Method Based on Particle Swarm Optimization and Probabilistic Neural Network Model. *International Conference on Network Computing and Information Security*, 288-292.
- Meng, Q., & Song, Y. (2012). Text Detection in Natural Scenes with Salient Region. *IAPR International Workshop on Document Analysis System*, 384-388.
- Pan, Y.F., Liu, C., & Hou, X. (2010). Fast scene text localization by learning-based filtering and verification. *IEEE International Conference on Image Processing*, 2269-2272.
- Shi, C., Xiao, B., Wang, C., & Zhang, Y. (2012). Graph-based Background Suppression For Scene Text Detection. *IAPR International Workshop on Document Analysis System*, 210-214.
- Shivakumara, P., & Tan, C. L. (2010). New Wavelet and Color Features for Text Detection in Video. *International Conference on Pattern Recognition*.
- Shivakumara, P., Phan, T. Q., & Tan, C. L. (2011). A Laplacian Approach to Multi-Oriented Text Detection in Video. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 33(2), 412-419.
- Wang, C., & Wang, H. (2010). Utilization of Temporal Continuity in Video Text Detection. *International Conference on MultiMedia and Information Technology*, 343-346.
- Wen-xu, W. (2011). Fault diagnosis of transformer based on probabilistic neural network. *International Conference on Intelligent Computation Technology and Automation*.
- Yang, H. (2012). An Improved Probabilistic Neural Network with GA Optimization. *International Conference on Intelligent Computation Technology and Automation*, 76-79.

Zhang, H., Zhao, K., Song, Z.Y., & Guo, J. (2013). Text extraction from natural scene image: A survey. *Neurocomputing*, 122, 310-323.

Zhao, H., Liu, C., Wang, H., & Li, C. (2010). Classifying ECoG Signals Using Probabilistic Neural Network. *WASE International Conference on Information Engineering*.

## BIOGRAFI PENULIS



tahun 2014.

**Endah Ekasanti Saputri.** Lahir pada tanggal 27 Oktober 1988 di Kab.Sintang, Kalimantan Barat. Memperoleh gelar Sarjana Teknik(S.T) dari fakultas Teknik Informatika, Institut Teknologi Telkom, Bandung (sekarang: Universitas Telkom) pada tahun 2010. Serta memperoleh gelar M.Kom dari Fakultas Ilmu Komputer, Universitas Dian Nuswantoro pada



bergerak di bidang pengembangan software. Minat penelitian pada bidang software engineering dan machine learning. Profesional member dari asosiasi ilmiah ACM, PMI dan IEEE Computer Society.

**Romi Satria Wahono.** Memperoleh Gelar B.Eng dan M.Eng pada bidang ilmu komputer di Saitama University, Japan, dan Ph.D pada bidang software engineering di Universiti Teknikal Malaysia Melaka. Menjadi pengajar dan peneliti di Fakultas Ilmu Komputer, Universitas Dian Nuswantoro. Merupakan pendiri dan CEO PT Brainmatics, sebuah perusahaan yang



Intelligence Control, Bremen Germany pada tahun 1999. Merupakan pengajar di fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang.

**Vincent Suhartono.** Memperoleh gelar Ing pada Information Technology and Broadcasting Technology dari Fachhochschule Bielefeld, Germany pada tahun 1979. Dan memperoleh gelar Dipl.-Ing pada Electronics Technology, Universitaet Bremen, Germany pada tahun 1986. Selain itu memperoleh Dr.-Ing dari fakultas Electrical Engineering and