

Implementasi Jaringan Saraf Tiruan untuk Pengenalan Karakter pada Studi Kasus Permasalahan SPOJ Hard Image Recognition

Cynthia Dewi Tejakusuma, Rully Soelaiman, dan Wijayanti Nurul Khotimah

Departemen Informatika, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember (ITS)
Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia

e-mail: cynthia.dewi15@mhs.if.its.ac.id, rully@if.its.ac.id, wijayanti@if.its.ac.id

Abstrak— Komputer semakin dirancang untuk memiliki kemampuan yang sama dengan manusia. Salah satu contoh kemampuan yang manusia kembangkan pada komputer adalah pengenalan karakter. Manusia dapat dengan mudah mengenali karakter pada citra yang mengalami transformasi, stretching, ataupun derau acak misalnya pada pembacaan nomor polisi kendaraan atau captcha. Manusia harus memberi algoritma pada komputer untuk proses pengenalan karakter agar dapat memiliki kemampuan yang diinginkan. Dengan kemampuan tersebut, komputer dapat membantu kegiatan manusia sehari-hari. Tugas akhir ini mengangkat salah satu subtopik pengenalan karakter yaitu klasifikasi digit numerikal. Penulis akan merancang penyelesaian permasalahan pengenalan karakter pada studi kasus permasalahan SPOJ Hard Image Recognition (HIR) dengan mengimplementasikan jaringan saraf tiruan menggunakan Bahasa Python dan C++. Data yang digunakan adalah citra pada situs daring penilaian SPOJ Hard Image Recognition (HIR). Penulis mendapatkan arsitektur jaringan saraf tiruan yang optimal, yaitu yang dapat mengenali karakter pada data SPOJ terbanyak atau mendapatkan nilai tertinggi pada SPOJ. Aplikasi yang dibuat berhasil mendapatkan nilai tertinggi pada SPOJ Hard Image Recognition (HIR), yaitu 32.

Kata Kunci— jaringan saraf tiruan, klasifikasi digit, pengenalan karakter

I. PENDAHULUAN

Komputer pada jaman sekarang semakin dirancang untuk memiliki kemampuan yang sama dengan manusia. Pengenalan karakter pada citra adalah salah satu kemampuan komputer yang sedang banyak dikembangkan oleh manusia dan sangat berguna dalam kehidupan manusia, diantaranya adalah untuk membaca nomor polisi kendaraan, membaca tulisan tangan manusia, dan jika ada proses yang ditambahkan, komputer dapat menterjemahkan kalimat dari suatu bahasa ke bahasa lainnya. Hal-hal tersebut dapat membantu pekerjaan manusia.

Topik Tugas Akhir ini mengacu pada permasalahan Online Judge SPOJ dengan kode HIR [1]. Permasalahan ini mengangkat subtopik pengenalan karakter yaitu klasifikasi digit numerical 0 sampai 9 pada citra biner. Tipe permasalahan ini adalah *challenge*, dimana penilaian akan dilakukan berdasarkan nilai. Semakin banyak program mengenali citra dengan benar, semakin tinggi nilai yang didapat. Pada permasalahan ini terdapat sekumpulan karakter 'X' dan '.' yang

merepresentasikan citra biner dari 6 angka yang didapat dari dataset SPOJ. 'X' merepresentasikan foreground atau warna hitam dan '.' merepresentasikan background atau warna putih dari citra tersebut. Terdapat citra yang mengalami beberapa perlakuan atau transformasi dan terdapat citra yang memiliki derau acak. Dari sekumpulan karakter ini, program yang telah dibuat akan melakukan proses pengenalan karakter sehingga keluaran dari program berupa 6 angka sesuai dengan citra.

Hasil tugas akhir ini diharapkan dapat memberi gambaran mengenai algoritma untuk menyelesaikan permasalahan di atas secara optimal dan diharapkan dapat memberikan kontribusi pada perkembangan ilmu pengetahuan dan teknologi informasi.

II. TINJAUAN PUSTAKA

A. Pengenalan Pola

Pengenalan pola atau dalam Bahasa Inggris adalah *pattern recognition* mencakup berbagai permasalahan pemrosesan informasi mulai dari pengenalan suara, klasifikasi karakter tulisan tangan, hingga deteksi kesalahan pada mesin[5]. Salah satu contoh permasalahan pengenalan pola adalah klasifikasi digit. Citra dari sebuah digit ditangkap oleh sebuah komputer diumpankan ke sebuah komputasi dan komputer mencari algoritma yang dapat membedakan setiap digitnya. Terdapat 10 kelas yang mewakili digit 0 sampai 9. Tujuan dari klasifikasi ini adalah untuk mengembangkan suatu algoritma yang akan menetapkan citra digit yang diwakili oleh vektor x , ke salah satu dari 10 kelas yang dilambangkan dengan C_k , dimana $k = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$, maka kelas C_1 menyatakan citra tersebut adalah citra digit 0, kelas C_2 menyatakan citra digit 1, dan seterusnya hingga kelas C_{10} menyatakan citra digit 9. Manusia harus mempersiapkan sejumlah besar contoh citra yang sesuai dengan digit 0 hingga digit 9 yang sudah diklasifikasikan oleh manusia. Citra-citra tersebut akan disebut sebagai suatu set data atau dapat disebut dengan sampel.

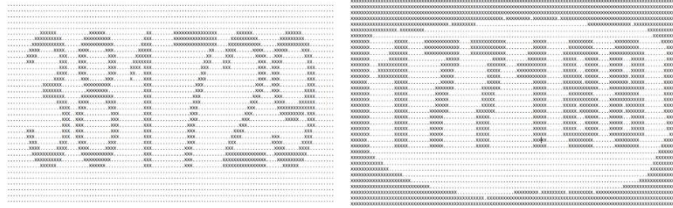
B. Citra Biner

Citra biner adalah citra yang hanya terdiri dari 2 kemungkinan nilai piksel, yaitu 0 dan 1, dimana pada *library opencv* Python 0 adalah hitam dan 1 adalah putih. Citra biner sering disebut dengan citra *black* dan *white*. Citra masukan program merupakan citra biner karena hanya terdiri dari 2 kemungkinan karakter yaitu 'X' dan '.' yang akan

direpresentasikan dengan nilai 1 dan 0. Gambar 1 adalah contoh merupakan contoh citra masukan yang berupa PNG. Gambar 2 adalah contoh citra masukan program C++ pada situs penilaian daring SPOJ Hard Image Recognition (HIR).

254839 381729 491881 177188 349010

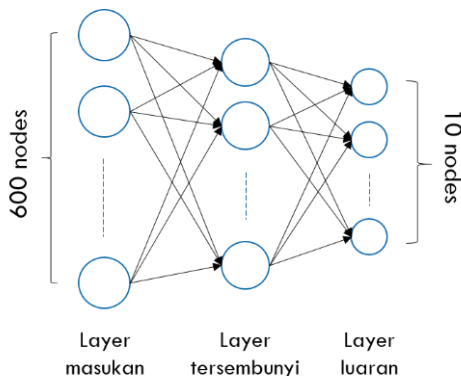
Gambar 1. Contoh Citra Masukan PNG



Gambar 2. Contoh Citra Masukan TXT

C. Jaringan Saraf Tiruan (JST)

Jaringan saraf tiruan atau dalam Bahasa Inggris disebut dengan *artificial neural network* adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan sistem saraf manusia[3]. JST merupakan salah satu algoritma untuk melakukan pengklasifikasian pada komputer. JST memiliki 3 lapisan, yaitu lapisan masukan, lapisan tersembunyi, dan lapisan luaran. JST bukan merupakan sebuah algoritma, namun sebuah *framework* untuk algoritma-algoritma *machine learning*. Jaringan saraf tiruan memiliki parameter utama yang dapat diubah nilainya, yaitu jumlah layer tersembunyi, fungsi aktivasi, fungsi optimasi, *learning rate*, dan *epoch*. Gambar 3 adalah struktur umum JST dan gambar 4 adalah ilustrasi perhitungan luaran pada JST.

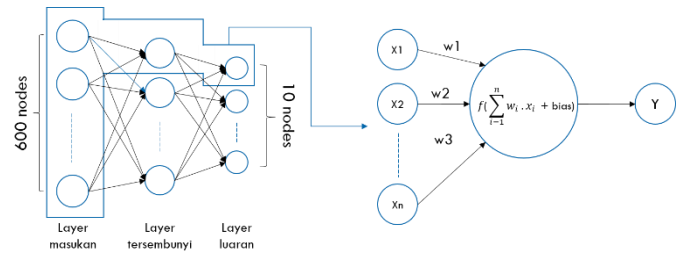


Gambar 3. Struktur JST

Di bidang citra, lapisan masukan merupakan fitur-fitur yang dimiliki oleh sebuah citra, yang dapat berupa nilai piksel-piksel citra ataupun fitur lainnya seperti luas, keliling, dan sebagainya. Lapisan tersembunyi adalah lapisan yang terletak antara lapisan masukan dan lapisan luaran yang menerima nilai yang sudah diproses dengan bobot pada masukan dan menghasilkan luaran yang teraktivasi. Lapisan luaran adalah lapisan terakhir yang ada pada jaringan saraf tiruan yang menghasilkan luaran dari program. Perhitungan luaran dilakukan dengan persamaan 1. Dimana Y adalah luaran, $f(\cdot)$ adalah fungsi aktivasi, w_i adalah bobot node ke- i , dan x_i adalah nilai node ke- i .

$$Y = f\left(\sum_{i=1}^n w_i \cdot x_i\right) + bias$$

Persamaan 1. Perhitungan luaran JST



Gambar 4. Ilustrasi Perhitungan Luarannya pada JST

Jaringan saraf tiruan melakukan pelatihan dengan cara:

1. Menghitung output
2. Membandingkan output yang diterima dengan target yang diinginkan
3. Menyesuaikan bobot dan mengulangi proses

Terdapat beberapa parameter arsitektur untuk membuat suatu JST yang nilainya akan diubah-ubah pada tahap uji coba untuk mendapatkan nilai tertinggi pada situs penilaian daring SPOJ. Komponen JST yang nilainya akan diubah-ubah pada tugas akhir ini adalah:

1. Jumlah Node pada Layer Tersembunyi
2. Fungsi Aktivasi
3. Fungsi Optimasi
4. Learning Rate
5. Jumlah Epoch

III. METODE PENYELESAIAN

Permasalahan SPOJ Hard Image Recognition (HIR) dapat diselesaikan dengan metode heuristik. Secara umum, penyelesaian permasalahan ini menggunakan jaringan saraf tiruan atau *artificial neural network*. Pada [6], JST digunakan untuk mengenali angka dengan dataset MNIST[7]. Penyelesaian akan dilakukan dengan menggunakan 2 program, yaitu dengan bahasa pemrograman Python untuk pemanggilan fungsi pembuatan dan pelatihan jaringan saraf tiruan dan C++ untuk pengujian jaringan saraf tiruan dengan citra-citra masukan dari SPOJ.

Bagian ini menjelaskan dataset yang digunakan pada Tugas Akhir ini dan alur kerja sistem serta metode yang digunakan. Alur kerja meliputi persiapan dataset, pemanggilan fungsi pelatihan JST, dan pengujian JST.

A. Persiapan Dataset

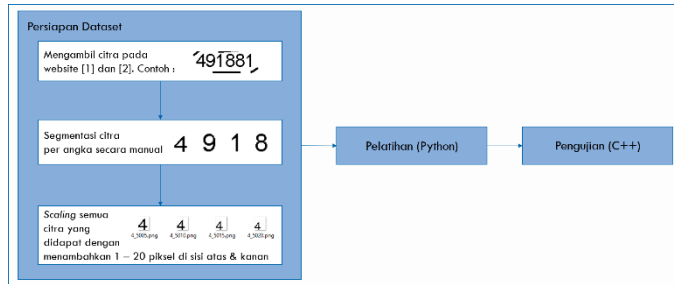
Langkah awal yang akan dilakukan adalah mempersiapkan citra pelatihan yang dapat diambil dari [1] dan [2] yang berjumlah 13 citra. Pada [1], terdapat 10 citra dalam bentuk PNG, sedangkan pada [2], terdapat 3 citra yang berupa berkas TXT yang berisi kumpulan karakter 'X' dan '.'. Satu citra terdiri dari 6 angka sehingga perlu dilakukan pemotongan per angka secara manual. Citra-citra yang didapat di-scaling dengan cara menambahkan 1 sampai 20 piksel pada sisi atas dan kanan untuk mendapatkan variasi citra. Jumlah citra pada dataset yang akan digunakan pada tahap pelatihan jaringan saraf tiruan adalah 1.596 citra. Proses persiapan dataset dapat dilihat pada Gambar 5.

B. Pemanggilan Fungsi Pembuatan dan Pelatihan JST

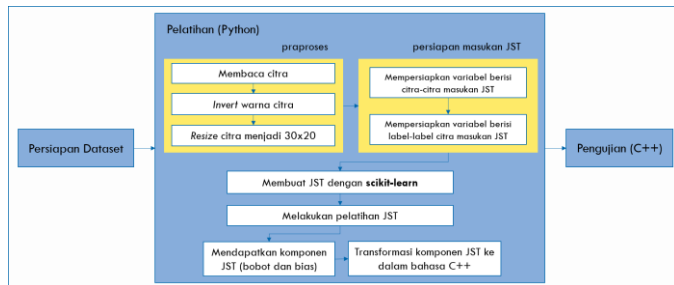
Pemanggilan fungsi pembuatan dan pelatihan jaringan saraf tiruan dilakukan dengan menggunakan bahasa pemrograman Python dengan menggunakan *library Scikit-learn*[4] fungsi MLPClassifier. Proses pemanggilan fungsi pembuatan dan pelatihan jaringan saraf tiruan dapat dilihat pada Gambar 6.

C. Pengujian JST

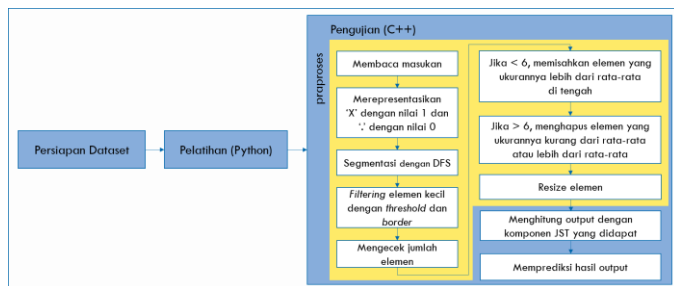
Pengujian jaringan dilakukan dalam sistem penilaian daring SPOJ dengan dataset pada sistem penilaian itu sendiri. Program pengujian JST ini dibuat menggunakan bahasa pemrograman C++. Proses pengujian jaringan saraf tiruan dapat dilihat pada Gambar 7.



Gambar 5. Proses Persiapan Dataset



Gambar 6. Proses Pelatihan Jaringan Saraf Tiruan



Gambar 7. Proses Pengujian Jaringan Saraf Tiruan

IV. UJI COBA DAN ANALISIS

A. Uji Coba Tahap Pemanggilan Fungsi Pembuatan dan Pelatihan JST

Uji coba dilakukan dengan melihat akurasi yang didapat dari hasil pengujian jaringan saraf tiruan dengan citra yang dilatih. Tujuan dari uji coba ini adalah menguji apakah jaringan saraf tiruan dapat mengenali citra yang dilatih menggunakan *library Scikit-learn* sebelum bobot dan bias jaringan dimasukkan ke dalam program pengujian JST dan dikirim ke situs penilaian daring SPOJ. Apabila jaringan belum bisa mengenali citra yang dilatih pada tahap pelatihan,

kemungkinan besar jaringan tidak akan bisa mengenali citra pengujian yang ada pada SPOJ Hard Image Recognition (HIR).

Tahap pemanggilan fungsi pembuatan dan pelatihan jaringan saraf tiruan menggunakan model MLP yang terdiri dari layer masukan, layer tersembunyi, layer luaran, fungsi aktivasi, fungsi optimasi, jumlah *epoch*, dan *learning rate*. Parameter lain menggunakan nilai *default* dari fungsi *library*. Jumlah *node* layer masukan adalah 600 sesuai dengan jumlah piksel pada citra, jumlah layer tersembunyi adalah 1, sedangkan jumlah *node* layer luaran adalah 10 sesuai dengan jumlah label yang tersedia. Kemudian dilakukan uji coba dengan parameter JST lainnya. Pengujian jaringan saraf tiruan dengan citra yang dilatih juga menggunakan *library Scikit-learn*. Dataset yang digunakan sejumlah 1.596 citra. Hasil uji coba tahap pemanggilan fungsi pembuatan dan pelatihan JST tiap komponen dengan citra pelatihan dapat dilihat pada Tabel 1-5.

Tabel 1. Uji Coba Learning Rate

Learning Rate	Akurasi (%)
0,005	99,93
0,001	99,93
0,05	100
0,01	100
0,5	100

Tabel 2. Uji Coba Fungsi Aktivasi

Fungsi Aktivasi	Akurasi (%)
Identity	100
Sigmoid	100
Tanh	100
ReLU	100

Tabel 3. Uji Coba Epoch

Epoch	Akurasi (%)
11.000	100
11.500	100
12.000	100
12.500	100
13.000	100

Tabel 4. Uji Coba Fungsi Optimasi

Epoch	Akurasi (%)
Adam	100
LBFGS	100
SGD	100

Tabel 5. Uji Coba Jumlah Node

Epoch	Akurasi (%)
20	100
25	100
30	100
35	100
40	100

Nilai akurasi tertinggi secara keseluruhan adalah 100%. Sehingga dapat disimpulkan bahwa jaringan merupakan jaringan yang optimal dalam memprediksi angka pada citra pelatihan. Namun, karena banyak perubahan parameter yang tidak berpengaruh secara signifikan terhadap akurasi pengujian

jaringan menggunakan citra yang dilatih, maka pengambilan keputusan untuk penggunaan nilai parameter akan dilakukan setelah melakukan uji coba pada situs daring SPOJ dengan mengirim program pengujian jaringan JST pada permasalahan SPOJ Hard Image Recognition (HIR).

B. Uji Coba Lokal Tahap Pengujian JST

Uji coba dilakukan dengan menjalankan program dan melihat apakah program dapat berjalan atau tidak ada error dan dapat mengeluarkan 6 angka hasil prediksi dari citra masukan program. Komponen dan nilai yang digunakan pada uji coba lokal program pengujian JST dapat dilihat pada Tabel 6.

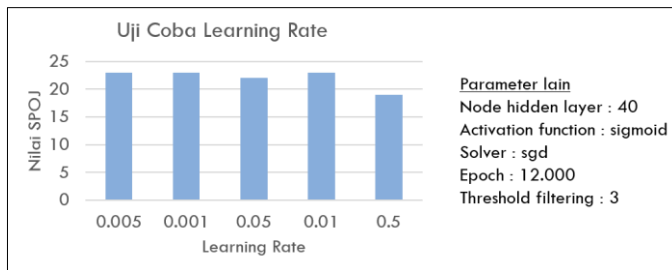
Tabel 6. Komponen yang Digunakan Uji Coba Lokal Tahap Pengujian JST

Nama Komponen	Nilai
Learning rate	0,01
Fungsi aktivasi	Sigmoid
Jumlah epoch	12.000
Fungsi optimasi	Adam
Jumlah node pada 1 layer tersembunyi	40
Threshold filtering	4

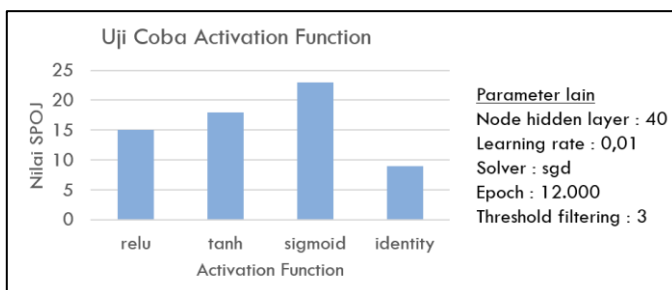
Evaluasi dilakukan dengan melihat nilai yang didapat dari hasil prediksi jaringan pada program pengujian JST secara lokal. Sama seperti pada situs penilaian daring SPOJ permasalahan Hard Image Recognition (HIR), nilai akan dinilai dengan jumlah citra yang benar terdeteksi oleh program. Dengan parameter yang ada pada Tabel 6, nilai yang didapat adalah 8 atau citra yang hasil prediksinya benar berjumlah 8 citra dari 13 contoh citra masukan pada [1] dan [2].

C. Uji Coba SPOJ Tahap Pengujian JST

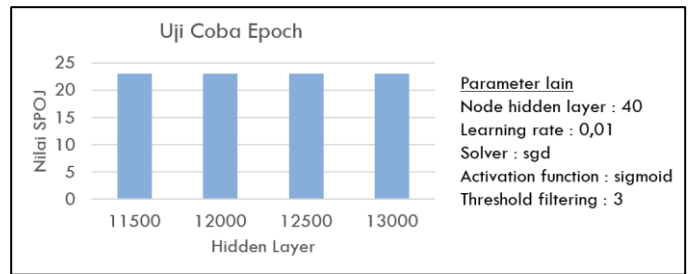
Uji coba dilakukan dengan mengubah-ubah nilai dari komponen yang ada pada Tabel 6 dan mengirim program pengujian JST ke situs SPOJ Hard Image Recognition (HIR). Nilai dari komponen yang diambil adalah yang mendapatkan nilai tertinggi. Hasil uji coba tahap pengujian JST tiap komponen pada SPOJ dapat dilihat pada Gambar 8-13.



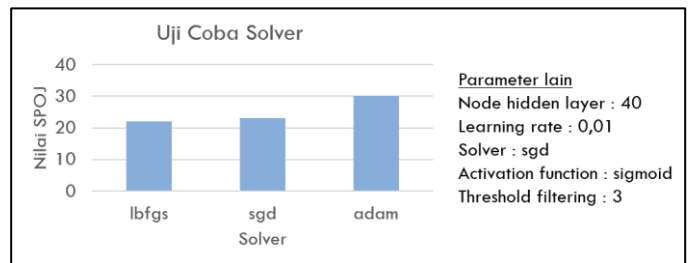
Gambar 8. Grafik Nilai SPOJ untuk Uji Coba Learning Rate



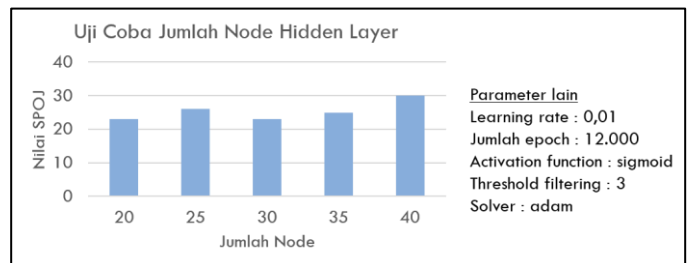
Gambar 9. Grafik Nilai SPOJ untuk Uji Coba Fungsi Aktivasi



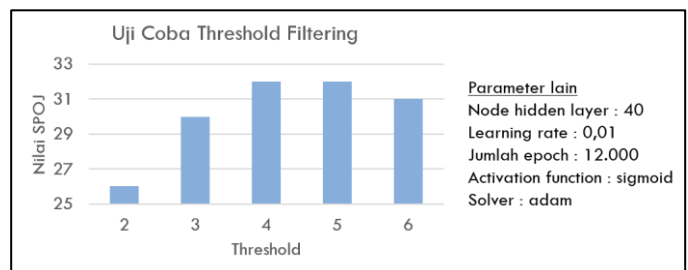
Gambar 10. Grafik Nilai SPOJ untuk Uji Coba Jumlah Epoch



Gambar 11. Grafik Nilai SPOJ untuk Uji Coba Fungsi Optimasi



Gambar 12. Grafik Nilai SPOJ untuk Uji Coba Jumlah Node pada Satu Hidden Layer



Gambar 13. Grafik Nilai SPOJ untuk Uji Coba Threshold Filtering

Dari hasil uji coba di atas, nilai tertinggi yang berhasil didapat pada situs penilaian daring SPOJ adalah 32. Komponen JST yang digunakan dan nilainya yang mendapatkan nilai SPOJ tertinggi dapat dilihat pada Tabel 7. Nilai dan peringkat pada SPOJ dapat dilihat pada Gambar 14 dan 15.

Tabel 7. Komponen yang Digunakan Uji Coba SPOJ dengan Nilai Tertinggi

Nama Komponen	Nilai
Learning rate	0,01
Fungsi aktivasi	Sigmoid
Jumlah epoch	12.000
Fungsi optimasi	Adam
Jumlah node pada 1 layer tersembunyi	40
Threshold filtering	4

22940892	2019-12-25 11:09:18	Hard Image Recognition	32	0.07	16M	CPP14
----------	---------------------	------------------------	----	------	-----	-------

Gambar 14. Hasil Pengiriman Program Pengujian JST ke Situs Penilaian Daring SPOJ Hard Image Recognition (HIR)

RANK	DATE	USER	RESULT	TIME	MEM	LANG
1	2019-12-25 09:29:59	CynthiaDewi	32	0.07	17M	CPP14
2	2019-08-24 11:44:09	Manish Kumar	2	0.00	2.2M	C
3	2019-05-01 13:22:59	ShayanOH	0	0.00	3.2M	CPP

Gambar 15. Peringkat Nilai pada Situs Penilaian Daring SPOJ Hard Image Recognition (HIR)

V. KESIMPULAN

Dari hasil uji coba yang telah dilakukan terhadap implementasi solusi untuk permasalahan SPOJ Hard Image Recognition, dapat diambil kesimpulan sebagai berikut:

1. Jaringan saraf tiruan yang dibuat merupakan jaringan yang optimal dalam pengujian terhadap dataset citra pelatihan, karena jaringan berhasil mencapai 100% dalam pengujian terhadap dataset citra pelatihan yang dipersiapkan.
2. Sistem yang diajukan secara lokal berhasil memprediksi 8 dari 13 contoh citra masukan dari [1] dan [2] dengan benar.
3. Sistem yang diajukan berhasil memprediksi 32 citra masukan dengan benar pada situs penilaian daring SPOJ Hard Image Recognition (HIR) dengan beberapa batasan dari situs, terutama batas ukuran kode sumber yang dikirim.
4. Sistem yang diajukan mendapatkan nilai tertinggi pada situs penilaian daring SPOJ Hard Image Recognition (HIR) yaitu 32 dengan waktu yang diperlukan adalah 0.07 detik, penggunaan memori adalah 16 MB, dan ukuran kode sumber yang dikirim adalah 139.264 B.
5. Sistem yang mendapatkan nilai tertinggi menggunakan dataset yang terdiri dari 1.569 citra dan memiliki komponen arsi tektur jumlah node layer masukan 600, jumlah node layer tersembunyi 40, jumlah epoch 12.000, learning rate 0,01, fungsi aktivasi sigmoid, fungsi optimasi adam, dan threshold filtering 4.
6. Sistem yang diajukan belum bisa memprediksi citra masukan dengan jumlah angka yang tersambung menjadi satu lebih dari 2.

UCAPAN TERIMA KASIH

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas pimpinan, penyertaan, dan karunia-Nya sehingga penulis dapat menyelesaikan penelitian ini. Penulis juga mengucapkan terima kasih kepada orang tua dan keluarga penulis, juga kepada Bapak Rully Soelaiman dan Ibu Wijayanti Nurul Khotimah selaku dosen pembimbing penulis dan kepada semua pihak yang telah memberikan dukungan baik secara langsung maupun tidak langsung selama penulis mengerjakan penelitian ini.

DAFTAR PUSTAKA

[1](). Hard Image Recognition, [Online]. Available: <https://www.spoj.com/problems/HIR>.

[2](). Hard Image Recognition Open Contest 2006, [Online]. Available: <http://discuss.spoj.com/t/hard-image-recognition/663>.

[3]Gurney, K. (1997), An Introduction to Neural Networks London: Routledge. ISBN 1-85728-673-1 (hardback) or ISBN 1-85728-503-4 (paperback)

[4] (). Konsep Depth First Search (DFS), [Online]. Available: <https://indonesia.hackerearth.com/konsep-depth-first-search-dfs/>.

[5]C. M. Bishop, Neural Networks for Pattern Recognition. New York, NY, USA: Oxford University Press, Inc., 1995, ISBN:0198538642.

[6]K. T. Islam, G. Muftaba, R. G. Raj, and H. F. Nweke, "Handwritten digits recognition with artificial neural network", in 2017 International Conference on Engineering Technology and Technopreneurship (ICE2T), 2017, pp. 1–4. DOI: 10.1109/ICE2T.2017.8215993.

[7] Y. LeCun, C. Cortes, and C. Burges, "MNIST handwritten digit database, 1998," URL <http://www.research.att.com/~yann/ocr/mnist>, 2016.

[8] R. Jain, R. Kasturi, and B. G. Schunck, Machine Vision. Jan. 1995, ISBN: 978-0-07-032018-5.

[9] T. M. Mitchell, Machine Learning, 1st ed. New York, NY, USA: McGraw-Hill, Inc., 1997, ISBN: 0070428077, 9780070428072.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python", Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.