

## Pengenalan Pola Huruf Hijaiyah Khat Kufi dengan Metode Deteksi Tepi Sobel Berbasis Jaringan Syaraf Tiruan Backpropagation

Irvan Faturrahman<sup>1</sup>, Arini<sup>2</sup>, Fitri Mintarsih<sup>3</sup>

<sup>1,2,3</sup>Program Studi Teknik Informatika, Fakultas Sains dan Teknologi  
Universitas Islam Negeri Syarif Hidayatullah Jakarta  
irvanfaturr1@gmail.com, arini@uinjkt.ac.id, mintarsihf@gmail.com

### ABSTRAK

Khat kufi memiliki bentuk huruf hijaiyah yang unik berbentuk kotak. Banyak penelitian yang membahas pengenalan huruf hijaiyah namun untuk spesifik khat belum ada. Pada penelitian ini penulis melakukan simulasi pengenalan pola huruf hijaiyah khat kufi menggunakan deteksi tepi *sobel* dan jaringan syaraf tiruan *backpropagation* dengan menggunakan parameter uji *learning rate* dan *epoch*. Simulasi dilakukan 28 target huruf hijaiyah dengan *learning rate* 0.01, 0.05, 0.1, 0.5, dan *epoch* 25, 1000, 3000, 5000, 10000. Akurasi terbaik didapatkan pada *learning rate* 0.01 dan *epoch* 10000 yaitu 100%. Penelitian ini dapat dikembangkan menggunakan deteksi tepi *canny*, *prewitt*, atau *robert* serta JST *LVQ*, *ADALINE*, atau *RBF*.

**Kata Kunci:** *Khat Kufi, Deteksi Tepi Sobel, Jaringan Syaraf Tiruan Backpropagation, Simulasi*

### ABSTRACT

Khat kufi has a unique hijaiyah shape that is square in shape. Much of the research that discusses the introduction of the hijaiyah letters but for the specifics khat does not yet exist. In this study, the author performs a simulation of hijaiyah khat kufi pattern recognition using sobel edge detection and artificial neural network backpropagation using learning rate test and epoch parameters. The simulation has been done on 28 target letters hijaiyah with learning rate 0.01, 0.05, 0.1, 0.5, and epoch 25, 1000, 3000, 5000, 10000. The best accuracy obtained at learning rate 0.01 and epoch 10000 is 100%. This research can be developed using canny edge detection, prewitt, or robert and also JST LVQ, ADALINE, or RBF.

**Keywords:** *Khat Kufi, Detection Of Sobel Edge, Backpropagation Artificial Neural Networks, Simulation*

DOI : 10.15408/jti.v11i1.6262

## I. PENDAHULUAN

### 1.1 Latar Belakang

Seni kaligrafi yang merupakan kebesaran seni Islam, lahir di tengah-tengah dunia arsitektur dengan segar bugar. Ini dapat dibuktikan pada aneka ragam hiasan kaligrafi yang memenuhi masjid-masjid dan bangunan-bangunan lainnya, yang ditumpahkan dalam paduan ayat-ayat al-Qur'an yang mulia, hadits-hadits atau kata-kata hikmat para ulama bijaksana. Kaligrafi sendiri mempunyai jenisnya masing-masing, salah satunya adalah khat kufi. Khat kufi merupakan khat yang paling tua di antara teman-temannya. Selain karena usianya, kaligrafi ini terkenal dengan bentuknya yang kotak-kotak. Karenanya, kaligrafi ini cukup mudah digoreskan jika ada mistar di sisi kita. Kufi merupakan salah satu khat paling populer untuk dekorasi masjid. Karena bentuknya yang kotak-kotak khat kufi sangat mudah dibedakan dalam penulisan dengan khat lainnya [1].

Kaligrafi sendiri biasanya ditulis tangan oleh seseorang yang bisa disebut dengan khatat. Setiap khatat mempunyai teknik tersendiri dalam menuliskan kaligrafi. Namun untuk dapat membaca sebuah kaligrafi seseorang perlu memahami bentuk dari masing-masing huruf hijaiyah terlebih dahulu. Menurut Ketua UKM Lemka UIN Syarif Hidayatullah periode 2014-2016, Mochammad Sholeh, walaupun seseorang telah mengetahui bagaimana bentuk dan pola dari huruf hijaiyah yang ditulis dengan gaya biasa seperti yang ada dalam al-Qur'an sekarang ini, tetap saja akan sulit untuk membaca sebuah tulisan kaligrafi. Oleh karena itu kita harus memahami betul bagaimana bentuk dan pola huruf hijaiyah dengan spesifik khat, disini peneliti menggunakan khat kufi. Sebab jika tidak bisa mengenali huruf hijaiyah maka akan sulit untuk membacanya. Untuk khat kufi sendiri memiliki bentuk huruf hijaiyah yang hampir sama dan unik satu dengan yang lainnya. Biasanya orang hanya mengira-ngira huruf hijaiyah tersebut. Sehingga perlu suatu program yang dapat mengenali huruf hijaiyah dengan khat kufi. Program tersebut difungsikan untuk mengenali karakter huruf hijaiyah dengan khat kufi yang kemudian memberikan solusi bagi mesin untuk dapat mengenalinya.

Pada penelitian ini peneliti melakukan pengenalan pola huruf hijaiyah khat kufi dengan menggunakan jaringan syaraf tiruan algoritma *backpropagation*. Peneliti menggunakan algoritma *backpropagation* karena algoritma

*backpropagation* termasuk ke dalam lapisan terawasi (*supervised*) yang mana terdapat sepasang data (masukan-target-keluaran) yang dipakai untuk melatih bobot yang diinginkan, dan algoritma *backpropagation* mampu mengenali pola yang akan digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang serupa (tapi tidak sama) dengan pola yang dipakai selama pelatihan [2].

Peneliti juga menggunakan teknik operasi pengolahan citra, yaitu deteksi tepi. Deteksi tepi sangat penting dalam pengolahan citra karena pendeteksian tepi melingkupi informasi di dalam citra. Dimana, tepi memberikan batas-batas objek citra. Tujuan operasi pendeteksian tepi adalah untuk meningkatkan penampakan garis batas suatu daerah atau objek di dalam citra. Ada banyak metode pendeteksian tepi, salah satunya adalah *sobel*. *Sobel* merupakan salah satu metode deteksi tepi yang memiliki kelebihan yaitu kemampuan untuk mengurangi *noise* sebelum melakukan perhitungan deteksi [3].

Sebelumnya telah banyak penelitian mengenai pengolahan citra pada pengenalan pola menggunakan jaringan syaraf tiruan *backpropagation* diantaranya penelitian yang dilakukan oleh [4], [5], [6], dan [7]. Semua penelitian tersebut melakukan pengenalan pola huruf seperti huruf hijaiyah dan huruf aksara Lampung serta identifikasi jenis kaligrafi yang memberikan hasil cukup optimal dalam pengenalan pola. Namun untuk penelitian mengenai pengenalan pola huruf hijaiyah dengan spesifik khat dengan kombinasi jaringan syaraf tiruan *backpropagation* dan deteksi tepi *sobel* belum pernah dilakukan sebelumnya sehingga peneliti tertarik untuk melakukan penelitian mengenai pengenalan pola huruf hijaiyah dengan khat kufi.

Berdasarkan latar belakang tersebut peneliti ingin melakukan sebuah pengenalan pola huruf hijaiyah khat kufi dan menjadikannya sebagai bahan kajian yang dituangkan dalam bentuk penelitian dengan judul "Pengenalan Pola Huruf Hijaiyah Khat Kufi dengan Metode Deteksi Tepi *Sobel* Berbasis Jaringan Syaraf Tiruan *Backpropagation*".

### 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya, rumusan masalah yang akan diangkat dalam penelitian ini adalah bagaimana menerapkan metode jaringan syaraf

tiruan *backpropagation* dan deteksi tepi *sobel* untuk mengenali pola huruf hijaiyah khat kufi.

### 1.3 Batasan Masalah

Pada penelitian ini untuk lebih memusatkan masalah yang ada agar tidak menyimpang dari topik maka batasan masalah hanya akan membahas:

1. Penelitian ini hanya membahas pengenalan pola satu jenis khat yaitu khat kufi.
2. Citra input berupa 28 huruf hijaiyah khat kufi dua dimensi hasil *scan* dan jenis *font shafa* menggunakan *software Nonosoft Khot* dengan posisi citra tegak dan arah penulisan horisontal.
3. Pengolahan citra yang digunakan dalam penelitian adalah *scaling*, *grayscale*, deteksi tepi *sobel*, *thresholding*, dan dilasi.
4. Proses pengenalan pola menggunakan jaringan syaraf tiruan *backpropagation* dengan menggunakan skenario dari nilai *learning rate* dan jumlah *epoch* sebagai parameter pelatihan.
5. Dalam penelitian ini perangkat keras yang digunakan hanya berupa laptop sebagai media untuk perancangan *software* Matlab dan pengujian sistem.
6. *Output* dari penelitian ini menghasilkan simulasi pengenalan pola huruf hijaiyah khat kufi.

## II. LANDASAN TEORI

### 2.1 Khat Kufi

Kufi termasuk tulisan paling dominan pada zaman dahulu. Ia dibuat setelah berdirinya dua kota muslim yaitu Basrah dan Kufah pada dekade kedua era Islam sekitar abad ke-8 Masehi. Ia memiliki bentuk huruf yang proporsional kaku dan persegi. Dari kata Kufah maka tulisan ini dikenal dengan Kufi. Ciri pokok tulisan kufi sangat jelas, yakni berukuran seimbang yang spesifik dengan sifat bersudut-sudut atau persegi menyolok, memiliki sapuan-sapuan garis vertikal pendek dan garis-garis horizontal yang memanjang dalam ukuran sama lebar. Akan tampak bahwa tulisan berbentuk empat persegi panjang. Dalam gaya hias atau iluminasi, ukuran tersebut terkadang tidak mengikat betul. Misalnya, pada sapuan garis vertikal yang dibikin panjang-panjang melebihi garis-garis horisontalnya. Namun tetap harus ditekankan, bahwa tulisan kufi adalah tulisan bersiku-siku [1].

### 2.2 Pengenalan Pola

Secara umum pengenalan pola adalah suatu ilmu untuk mengklasifikasikan atau menggambarkan sesuatu berdasarkan pengukuran kuantitatif fitur (ciri) atau sifat utama dari suatu objek. Pola bisa merupakan kumpulan hasil pengukuran atau pemantauan dan bisa dinyatakan dalam notasi vektor atau matriks [3].

### 2.3 Deteksi Tepi Sobel

Operator sobel adalah salah satu operator yang menghindari adanya perhitungan gradien di titik interpolasi. Operator sobel menggunakan 2 matriks berukuran 3x3 yaitu matriks  $G_x$  dan  $G_y$ . Kedua matriks tersebut digunakan untuk menghitung perbedaan warna pada piksel yang sedang dihitung dengan piksel di sekitarnya secara horisontal dan vertikal. Matriks  $G_x$  digunakan untuk menghitung secara horisontal dan matriks  $G_y$  secara vertikal [8].

-1	0	+1
-2	0	+2
-1	0	+1

$G_x$

+1	+2	+1
0	0	0
-1	-2	-1

$G_y$

Gambar 1. Matriks metode Sobel

### 2.4 Jaringan Syaraf Tiruan

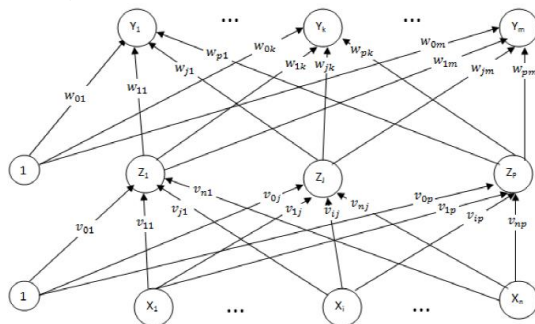
Menurut Fausett dalam [9], jaringan syaraf tiruan adalah generalisasi dari pemodelan syaraf biologi dengan asumsi-asumsi antara lain:

1. Pemrosesan informasi terletak pada sejumlah komponen yang dinamakan *neuron*.
2. Sinyal merambat antara satu *neuron* ke *neuron-neuron* lainnya melalui jalur penghubung.
3. Tiap jalur penghubung memiliki bobot dan mengalikan besar nilai sinyal yang masuk (jenis *neuron* tertentu).
4. Tiap neuron menerapkan fungsi aktivasi (biasanya *nonlinear*) yang menjumlahkan semua masukan untuk menentukan sinyal keluarannya.
5. Tiap jaringan ditentukan oleh arsitektur pola jaringan, bobot pada koneksi dan fungsi aktivasi.

**2.5 Jaringan Syaraf Tiruan *Backpropagation***

*Backpropagation* merupakan salah satu algoritma pelatihan terawasi. Algoritma ini biasa digunakan untuk mengubah bobot-bobot yang terhubung dengan *neuron-neuron* yang ada pada lapisan tersembunyinya. *Backpropagation* menggunakan *error output* untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*).

*Backpropagation* memiliki beberapa unit yang ada dalam satu atau lebih *layer* tersembunyi. Gambar 2 di bawah ini merupakan arsitektur *backpropagation* dengan n buah masukan (ditambah sebuah bias), sebuah *layer* tersembunyi yang terdiri dari p unit (ditambah sebuah bias), serta m buah unit keluaran [2].



Gambar 2. Proses *backpropagation*

Algoritma *backpropagation* untuk pelatihan dengan jaringan satu *layer* tersembunyi (menggunakan fungsi aktivasi sigmoid biner) adalah sebagai berikut:

**Langkah 0:** Inisialisasi semua bobot dengan bilangan acak kecil.

**Langkah 1:** Jika kondisi penghentian belum terpenuhi, lakukan langkah 2-9.

**Fase I:** Propagasi maju

**Langkah 2:** Untuk setiap pasang data pelatihan, lakukan langkah 3-8.

**Langkah 3:** Tiap unit masukan menerima sinyal dan meneruskannya ke unit tersembunyi di atasnya.

**Langkah 4:** Hitung semua keluaran unit tersembunyi  $z_j$

( $j = 1, 2, \dots, p$ ).

$$z_{net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ij}$$

$$z_j = f(z_{net_j}) = \frac{1}{1 + e^{-y_{net_j}}}$$

**Langkah 5:** Hitung semua keluaran jaringan di unit  $y_k$ .

( $k = 1, 2, \dots, m$ ).

$$y_{net_k} = w_{k0} + \sum_{j=1}^p z_j w_{kj}$$

$$y_k = f(y_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}}$$

**Fase II:** Propagasi Mundur

**Langkah 6:** Hitung faktor  $\delta$  unit keluaran berdasarkan kesalahan di setiap unit keluaran  $y_k$  ( $k = 1, 2, \dots, m$ ).

$$\delta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k)$$

$\delta_k$  merupakan unit kesalahan yang akan dipakai dalam perubahan bobot *layer* dibawahnya (langkah 7).

Hitung suku perubahan bobot  $w_{kj}$  (yang akan dipakai nanti untuk merubah bobot  $w_{kj}$ ) dengan laju percepatan  $\alpha$ .

$$\Delta w_{kj} = \alpha \delta_k z_j; k = 1, 2, \dots, n; j = 0, 1, \dots, p$$

**Langkah 7:** Hitung faktor  $\delta$  unit tersembunyi berdasarkan kesalahan di setiap unit tersembunyi  $z_j$  ( $j = 1, 2, \dots, p$ ).

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj}$$

Faktor  $\delta$  unit tersembunyi:

$$\delta_j = \delta_{net_j} f'(z_{net_j}) = \delta_{net_j} z_j (1 - z_j)$$

Hitung suku perubahan bobot  $v_{ij}$  (yang akan dipakai nanti untuk merubah bobot  $v_{ij}$ ).

$$\Delta v_{ij} = \alpha \delta_j x_i; j = 1, 2, \dots, p; i = 0, 1, \dots, n$$

**Fase III:** Perubahan Bobot

**Langkah 8:** Hitung semua perubahan bobot.

Perubahan bobot garis yang menuju ke unit keluaran

$$w_{kj}(\text{baru}) = w_{kj}(\text{lama}) + \Delta w_{kj}$$

( $k = 1, 2, \dots, m; j = 0, 1, \dots, p$ )

Perubahan bobot garis yang menuju ke unit tersembunyi

$$v_{ji}(\text{baru}) = v_{ji}(\text{lama}) + \Delta v_{ji}$$

( $j = 1, 2, \dots, p; i = 0, 1, \dots, n$ )

**2.6 Metode Simulasi**

Konsep sistem simulasi muncul dan dilaksanakan pada permulaan tahun 1950-an. Konsep ini muncul sebagai akibat dari terjadinya berbagai perubahan di dalam memandang persoalan, dimana suatu persoalan dianggap dapat diuraikan menurut bagian-bagian yang berinteraksi secara simultan. Perubahan-perubahan semacam ini secara nyata dapat diamati dalam percobaan. Sistem simulasi memberikan hasil yang layak (*feasible*) pada EDP, dimana hasilnya dapat diperoleh dengan cepat. Simulasi juga memberikan kemungkinan untuk mengerjakan seluruh bagian dalam sistem analisis yang sebenarnya merupakan persoalan yang kompleks yang harus dikerjakan dengan analisis [10]. Tahapan simulasi sebagai berikut.

1. Formulasi Permasalahan (*Problem Formulation*)
2. Pemodelan Formulasi (*Modelling Formulation*)
3. Persiapan Pengambilan Data (*Data Preparation*)
4. Penulisan Program (*Write Program*)
5. Verifikasi dan Validasi (*Verification and Validation*)
6. Desain Eksperimen (*Experimental Design*)
7. Perencanaan Taktis (*Tactical Planning*)
8. Pelaksanaan Percobaan (*Experiment Done*)
9. Model Terpakai (*Useful Model*)
10. Dokumentasi (*Documentation*)

### III. METODOLOGI PENELITIAN

#### 3.1 Metode Pengumpulan Data

Tahapan pengumpulan data yang digunakan dalam penelitian adalah sebagai berikut:

1. Studi Pustaka dan Literatur Sejenis
2. Wawancara

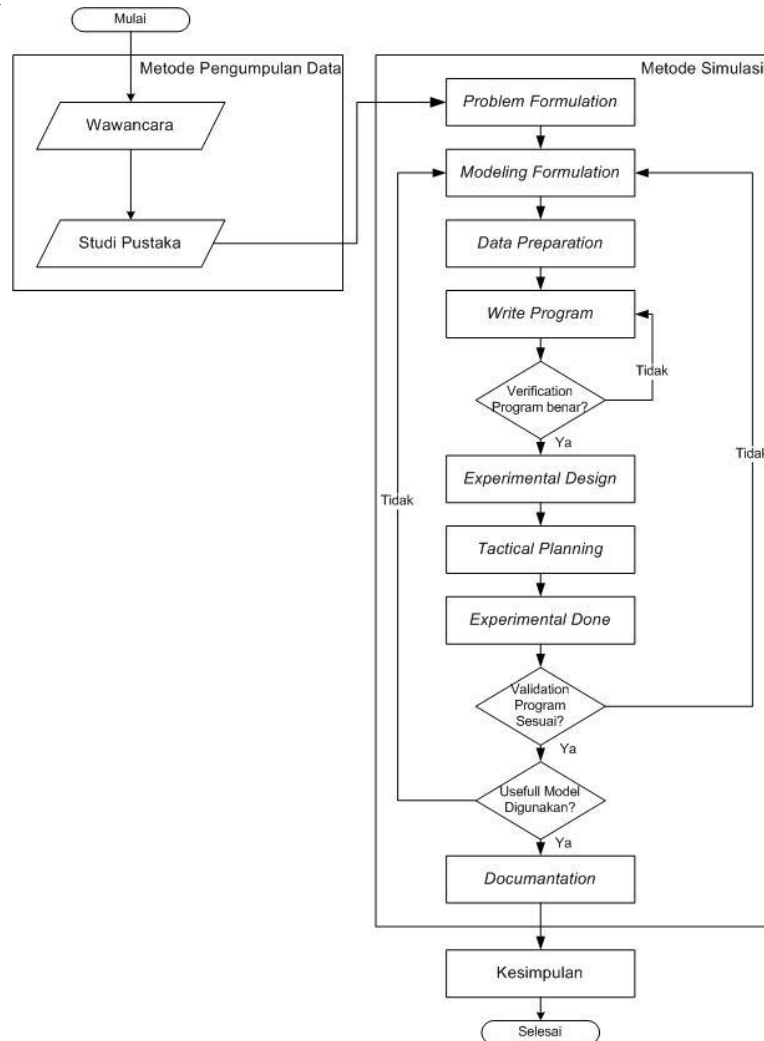
#### 3.2 Metode Simulasi

1. *Problem Formulation*  
Pada tahap ini penulis mengumpulkan data dan menganalisis masalah dari hasil studi pustaka dan wawancara.
2. *Modelling Formulation*  
Setelah permasalahan dirumuskan, selanjutnya dibuat pemodelan yang akan digunakan untuk melakukan simulasi pelatihan dan identifikasi pola huruf hijaiyah menggunakan jaringan syaraf tiruan *backpropagation* dengan diawali pra-proses citra menggunakan deteksi tepi *sobel*.
3. *Data Preparation*  
Langkah selanjutnya yaitu menentukan *input* yang akan di proses dan *output* yang akan didapat.
4. *Write Program*  
Tahap selanjutnya yaitu pengkodean. Pada tahap ini, aplikasi yang akan digunakan untuk simulasi dibuat dengan MATLAB (*Matrix Laboratory*) sesuai pemodelan yang dibuat pada tahap kedua.
5. *Verification and Validation*  
Selanjutnya memeriksa kesesuaian program dengan simulasi yang diinginkan. Selain itu, dilakukan *debugging* dan perbaikan berupa pengeditan pada program. Jika program yang dibuat dapat berjalan di

komputer, maka akan dilanjutkan dengan tahap berikutnya.

6. *Experimental Design*  
Desain eksperimen menjelaskan bagaimana data input didapatkan untuk pelaksanaan percobaan pada model. Dalam hal ini sampel huruf hijaiyah khat kufi.
7. *Tactical Planning*  
Dalam perencanaan ini dibutuhkan pemeriksaan kembali terhadap hasil dari tahap desain eksperimen. Pemeriksaan dilakukan dengan menyesuaikan sampel yang telah disediakan dengan hasil sampel huruf hijaiyah yang seharusnya.
8. *Experiment Done*  
Penulis melakukan percobaan dengan memasukkan data input pada program yang telah dibuat dan lolos tahap verifikasi. Jika hasil percobaan sesuai kebutuhan penulis, maka program tersebut dapat digunakan sebagai model.
9. *Useful Model*  
Model yang telah selesai dilakukan percobaan untuk memeriksa kesesuaian program yang dibuat dengan model yang diinginkan, selanjutnya digunakan untuk mendapatkan data yang digunakan penulis untuk dianalisis tingkat akurasi.
10. *Documentation*  
Langkah ini dilakukan dengan merekam program yang telah dibuat menjadi model simulasi dan mendokumentasikan hasil analisis ke dalam sebuah laporan.

### 3.3 Kerangka Berpikir



Gambar 3. Kerangka pemikiran

## IV. IMPLEMENTASI SIMULASI DAN EKSPERIMEN

### 4.1 Problem Formulation

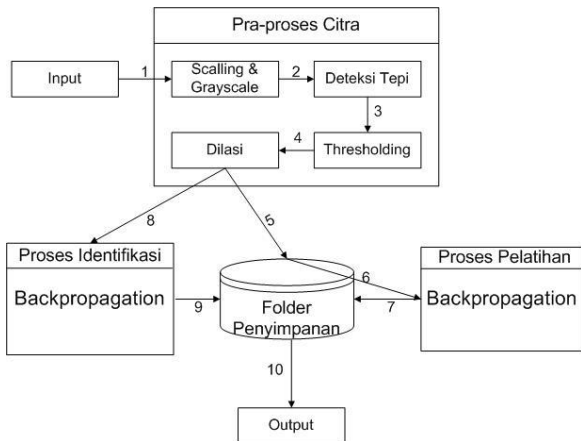
Berdasarkan permasalahan yang telah dipaparkan sebelumnya, penulis menerapkan gabungan dari salah satu metode pengolahan citra dan jaringan syaraf tiruan yaitu, deteksi tepi *sobel* dan jaringan syaraf tiruan *backpropagation*. Penerapan ini diharapkan dapat menghasilkan proses pembelajaran pada sistem menggunakan jaringan syaraf tiruan *backpropagation* yang nantinya akan memberikan hasil pengenalan pola huruf hijaiyah khatkufi.

Pada penelitian ini, masalah yang harus diselesaikan adalah bagaimana menerapkan metode jaringan syaraf tiruan *backpropagation* dan deteksi tepi *sobel* untuk mengenali pola huruf hijaiyah khat kufi. Penulis akan

melakukan simulasi pada sistem yang telah dibuat dengan parameter pengaruh hasil jaringan, *learning rate*, maksimal *epoch* terhadap akurasi pengenalan.

### 4.2 Modelling Formulation

Simulasi sistem dibuat berdasarkan pemodelan yang dirancang pada tahap ini. Sistem dibuat untuk dapat menjalankan tahap pelatihan dan identifikasi dari metode jaringan syaraf tiruan *backpropagation*.



Gambar 4. Gambaran umum pengenalan pola huruf

**4.3 Data Preparation**

Simulasi berjalan dengan skenario simulasi berikut:

Tabel 1. Skenario simulasi

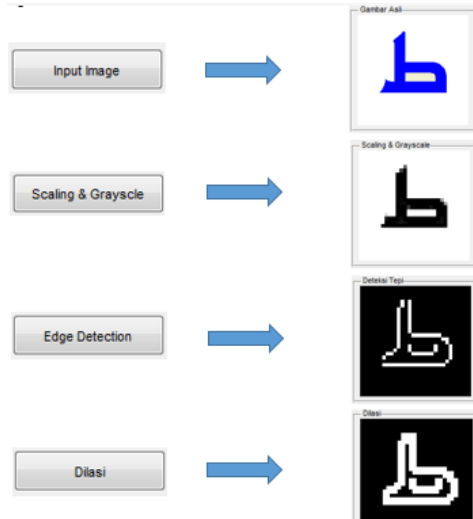
Indikator Penilaian	Skenario	Jumlah Epoch	Nilai Learning Rate	Keterangan
Pengaruh Nilai Learning Rate	LR1	25	0.01	Nilai Default
	LR2	25	0.05	
	LR3	25	0.1	
	LR4	25	0.5	
Pengaruh Jumlah Epoch	E1	1000	0.01	
	E2	3000	0.01	
	E3	5000	0.01	
	E4	10000	0.01	

**V. HASIL DAN PEMBAHASAN**

**5.1 Verification and Validation**

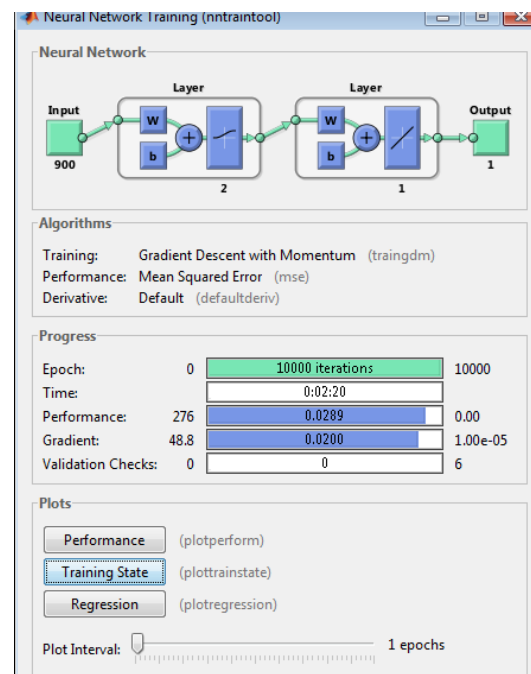
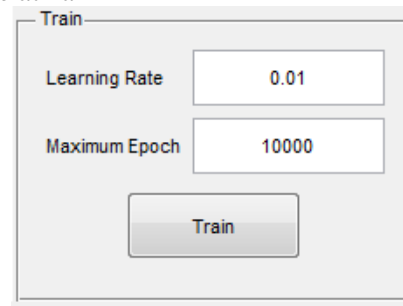
Verifikasi dan validasi dilakukan dengan memeriksa kesesuaian program yang telah dibuat dengan hasil dari *debugging* program pada komputer.

**1. Pra Proses Citra**



Gambar 5. Verifikasi pra proses citra

**2. Pelatihan**



Gambar 6. Verifikasi pelatihan

### 3. Identifikasi



Gambar 7. Verifikasi identifikasi

### 5.2 Experimental Design

Data simulasi yang digunakan berukuran 300x300 piksel menggunakan *tools Adobe Photoshop*. Data terbagi menjadi dua, yaitu data latih dan data uji. Data latih untuk simulasi ini terdiri dari dua sampel pada masing-masing huruf hijaiyah tersebut. Untuk data uji, citra hijaiyah khatkufi diambil dari hasil tulisan tangan anggota UKM Lemka UIN Syarif Hidayatullah Jakarta (Ahmad Kusairi) dan *font huruf shafa* pada *software Nonosoft Khot*. Data uji juga dibuat berukuran 300 x 300 piksel yang nantinya akan di *resize* oleh sistem pada tahap *scaling* hingga berukuran 30 x 30 piksel. Sama dengan data latih, data uji untuk 28 pola huruf hijaiyah terdiri dari dua sampel terhadap masing-masing huruf hijaiyah.

### 5.3 Tactical Planning

Simulasi dilakukan pada 28 huruf hijaiyah khat kufi. Masing-masing huruf hijaiyah dibutuhkan dua sampel sebagai data latih. Sehingga perhitungan jumlah data sampel yang dibutuhkan sebagai data latih dan data uji pada simulasi ini.

### 5.4 Experiment Done

Percobaan dilakukan setelah model dapat berfungsi sesuai dengan model yang telah dirancang. Pemeriksaan fungsi pada program yang telah berhasil di-*debug* dengan memasukkan jumlah *epoch* dan nilai *learning rate* untuk pelatihan dan mengidentifikasi citra yang dimasukkan sesuai hasil jaringan yang telah dilatih.

### 5.5 Useful Model

Pada tahap ini dilakukan skenario simulasi sesuai dengan *data preparation* yang telah diberikan sebelumnya kemudian hasilnya dirangkum dalam tahap dokumentasi.

### 5.6 Documentation

Setelah dilakukan pengujian berdasarkan skenario yang sebelumnya terhadap sampel-sampel citra huruf hijaiyah khat kufi, maka didapat hasil sebagai berikut.

Tabel 2 Tingkat akurasi hasil uji coba skenario

No	Skenario	Akurasi	No	Skenario	Akurasi
1	LR1	3,57%	5	E1	56,25%
2	LR2	3,57%	6	E2	81,25%
3	LR3	3,57%	7	E3	94,64%
4	LR4	3,57%	8	E4	100%

Hasil rangkuman pada Tabel 2 terlihat bahwa skenario E4 memiliki nilai akurasi tertinggi yaitu mencapai 100%. Pada tabel tersebut juga terlihat bahwa skenario LR1, LR2, LR3, dan LR4 memiliki akurasi paling kecil jika dibandingkan dengan skenario *epoch*. Hal itu disebabkan oleh beberapa faktor, diantaranya yaitu:

1. Pengaruh perbedaan masing-masing sampel  
Pada penelitian menggunakan dua sampel berbeda dengan masing-masing sampel diberikan satu sampel lain untuk melihat perbedaan hasil *backpropagation*-nya. Terlihat sedikit perbedaan pada sampel yang sama sehingga nilai *output* pelatihan dan *output* identifikasi hampir mendekati satu dengan yang lain
2. Pengaruh pengolahan  
Tahap *scaling* pada pengolahan citra (pre-proses) dan identifikasi akan menentukan akurasi sistem. Semakin besar ukuran citra huruf yang digunakan maka waktu komputasi sistem akan semakin lama. Namun jika terlalu kecil ukuran citra huruf maka matriks *sobel* di dalamnya akan bernilai hampir sama antara citra satu dengan yang lainnya. Sehingga diperlukan *resizing* yang tidak terlalu besar dan juga terlalu kecil. Dan skala *resize* yang digunakan dalam penelitian ini adalah 0.1.
3. Pengaruh parameter pelatihan  
Ada beberapa parameter-parameter jaringan syaraf tiruan algoritma *backpropagation* yang digunakan dalam penelitian ini yaitu:

```
- net.trainParam.lr =
  LearningRate;
```

Digunakan untuk menentukan laju pemahaman ( $\alpha$ ). Semakin besar laju pemahaman maka semakin cepat pelatihannya. Tetapi semakin besar



nilai laju pemahamannya maka algoritma menjadi semakin tidak stabil dan akurasi rendah.

- `net.trainParam.epochs = Epoch;`  
Digunakan untuk menentukan jumlah maksimum *epoch* pelatihan (iterasi). Pada penelitian ini semakin besar jumlah *epoch* maka proses pelatihan membutuhkan waktu yang lama tapi menghasilkan tingkat akurasi yang cukup tinggi. Namun sebaliknya semakin kecil jumlah *epoch* maka proses pelatihan membutuhkan waktu yang relatif singkat tapi menghasilkan tingkat akurasi yang rendah.
- `net.trainParam.goal = 0;`  
Digunakan untuk menentukan batas nilai MSE agar iterasi dihentikan. Proses pelatihan akan berhenti juga nilai  $MSE < \max MSE$  yang ditentukan atau jumlah *epoch* = nilai *epoch* yang ditentukan. *Default* nilai MSE adalah 0, dan pelatihan ini menggunakan  $MSE = 0$ . Nilai MSE yang didapatkan pada proses pelatihan terdapat pada Tabel 3 berikut ini.

Tabel 3 Nilai MSE Skenario

No.	Skenario	MSE
1	LR1	66.286
2	LR2	66.2565
3	LR3	65.2589
4	LR4	66.8746
5	E1	3.5116
6	E2	0.6476
7	E3	0.31099
8	E4	0.028944

## VI. PENUTUP

### 6.1 Kesimpulan

Berdasarkan rumusan masalah dan penjelasan yang sudah penulis jelaskan pada bab sebelumnya, maka penulis membuat kesimpulan bahwa huruf hijaiyah khat kufi dapat dikenali polanya pada sistem dengan mengambil nilai matriks hasil deteksi tepi dan dilasi melalui teknik pengolahan citra dan jaringan syaraf tiruan dengan menggunakan algoritma *backpropagation*.

Alur proses secara keseluruhan terlebih dahulu dilakukan pra-proses citra, melakukan pembelajaran dengan jaringan syaraf tiruan dan melakukan pengenalan pola. Tahapan pra-proses

citra yang dilakukan yaitu akuisisi citra, *scaling* dan *grayscale*, deteksi tepi, *thresholding*, dilasi dan selanjutnya matriks hasil dilasi disimpan di sebuah *folder* penyimpanan.

Pelatihan jaringan syaraf tiruan dengan algoritma *backpropagation* menggunakan 8 skenario pelatihan dengan paramater *epoch* 1000, 3000, 5000, dan 10000 serta *learning rate* 0.01, 0.05, 0.1, 0.5. Berdasarkan skenario pelatihan yang telah dilakukan didapatkan hasil maksimal pada skenario E4 (lihat Tabel 2) dengan tingkat akurasi 100%.

### 6.2 Saran

Penulis menyarankan untuk mengembangkan penelitian ini agar menjadi lebih baik dengan melakukan beberapa hal seperti simulasi peneltian yang dilakukan tidak hanya berbasis *dekstop*, tapi juga *smartphone*. Pengenalan pola tidak hanya untuk mengenali huruf tapi juga karakter huruf dan *output* yang diberikan berupa suara. Penelitian ini dapat dikembangkan dengan melakukan pengenalan pola khat lain. Juga tidak hanya huruf tapi juga mengenali kata atau kalimat. Kemudian dapat pula menggunakan teknik pengenalan pola lainnya seperti PCA, JST LVQ, JST ADALAIN, serta deteksi tepi lain seperti *canny*, *robert*, dan *prewitt*.

## DAFTAR PUSTAKA

- [1] Sirojuddin, D. 2007. *Seni Kaligrafi Islam*. Jakarta: Multi Kreasi Singgasana.
- [2] Siang, J.J. (2009). *Jaringan Syaraf Tiruan dan Pemrograman Menggunakan Matlab*. Yogyakarta: Andi.
- [3] Putra, Darma. 2010. *Pengolahan Citra Digital*. Yogyakarta: Andi.
- [4] Abed, Majida Ali dan Hamid Ali Abed Alasad. 2015. *High Accuracy Arabic Handwritten Characters Recognition Using Error Back Propagation Artificial Neural Networks*. University of Tikrit. Tikrit Iraq.
- [5] Kanta, Imam Anggara. 2013. *Pengenalan Pola Huruf Hijaiyah Tulisan Tangan Menggunakan Logika Fuzzy dengan Jaringan Syaraf Tiruan Backpropagation*. Universitas Muhammadiyah Surakarta. Surakarta.
- [6] Jas, Nurul Ainis. 2016. *Identifikasi Jenis-jenis Kaligrafi Islam dengan Menggunakan Teknik Pengolahan Citra dan Jaringan Syaraf Tiruan Algoritma*

- Backpropagation*. UIN syarif Hidayatullah. Jakarta.
- [7] Hara, Eliza. 2016. *Pengenalan Tulisan Tangan Aksara Lampung dengan Metode Deteksi Tepi (Canny) Berbasis Jaringan Syaraf Tiruan Backpropagation*. Universitas Negeri Lampung. Lampung.
- [8] Sutoyo, T. 2009. *Teori Pengolahan Citra Digital*. Yogyakarta: Andi.
- [9] Widodo, P dkk. 2013. *Penerapan Data Mining dengan Matlab*. Bandung: Rekayasa Sains.
- [10] Kakiay, T. 2004. *Pengantar Sistem Simulasi*. I. Yogyakarta: Andi.