



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk mengubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Pada penelitian ini, dibutuhkan beberapa tahap pelaksanaan agar metodologi dan perancangan sistem dapat berjalan dengan baik. Tahap-tahap tersebut yang dilaksanakan tersebut antara lain sebagai berikut.

1. Studi Literatur

Tahap ini bertujuan untuk mengumpulkan segala sumber pengetahuan yang dibutuhkan mulai dari jurnal ilmiah dari berbagai sumber seperti jurnal online yang didapatkan dari konferensi ataupun seminar ataupun ranah penelitian ilmiah sejenisnya yang membawa kepada arah ilmu pengetahuan yang dibutuhkan seperti mengumpulkan pemahaman khususnya pada penggunaan jaringan saraf tiruan maupun PCA.

2. Proses Perancangan

Tahap ini bertujuan untuk menentukan tujuan dari penelitian serta menentukan proses apa saja yang dibutuhkan selama penggunaannya, yang juga didasari dari berbagai literatur yang telah dipelajari selama proses tahap sebelumnya.

3. Implementasi

Pada tahap ini bertujuan untuk melakukan penyusunan aplikasi berdasarkan dengan proses perancangan aplikasi yang telah disusun dan

diimplementasikan sedemikian rupa berdasarkan apa yang telah dipelajari dan dan dirancang seperti yang telah disebutkan dalam tahap sebelumnya.

4. Uji Coba dan Evaluasi

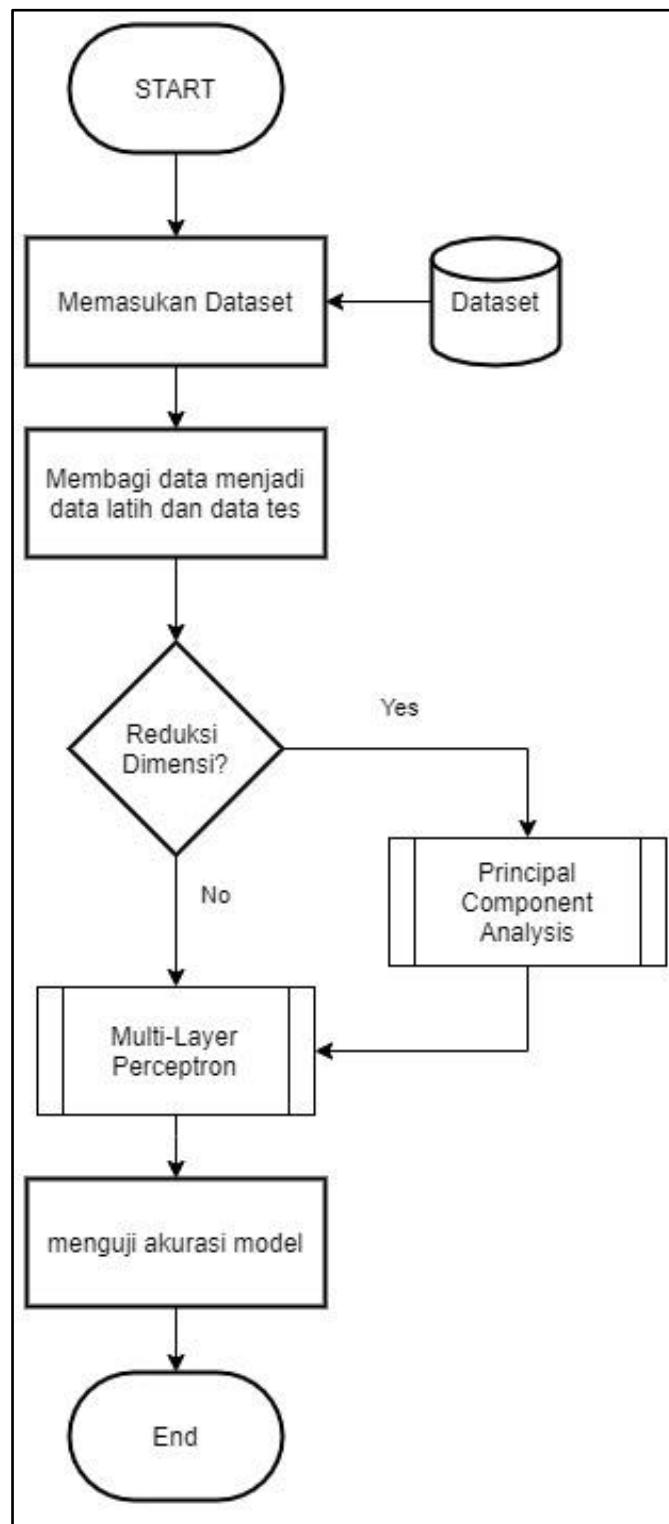
Pada tahap ini, dilakukan pengujian dari hasil penelitian yang didapat dari hasil program yang telah dirancang sebelumnya. Untuk mengetahui tingkat kesuksesan uji coba maka diperlukan evaluasi dengan menggunakan backpropagation untuk mengetahui tingkat akurasi.

5. Konsultasi dan Penulisan Laporan

Pada tahap ini, merupakan masa penyusunan laporan berdasarkan hasil penelitian yang telah dirancang dan dibangun pada tahap – tahap sebelumnya. Dalam penyusunannya, dibutuhkan pula konsultasi dari pembimbing untuk dapat membantu terlaksananya dengan baik laporan yang telah dirancang sebelumnya.

3.2 Perancangan Sistem

Rancangan sistem yang akan dilakukan selama proses implementasi dan uji coba nantinya yang telah didukung oleh beragam sumber dan *library* yang berguna untuk mempercepat proses dan memberikan hasil yang semakin baik. Berikut adalah alur rancangan sistem yang dibuat dalam bentuk flowchart guna memberikan gambaran proses utama seperti yang ada pada Gambar 3.1.

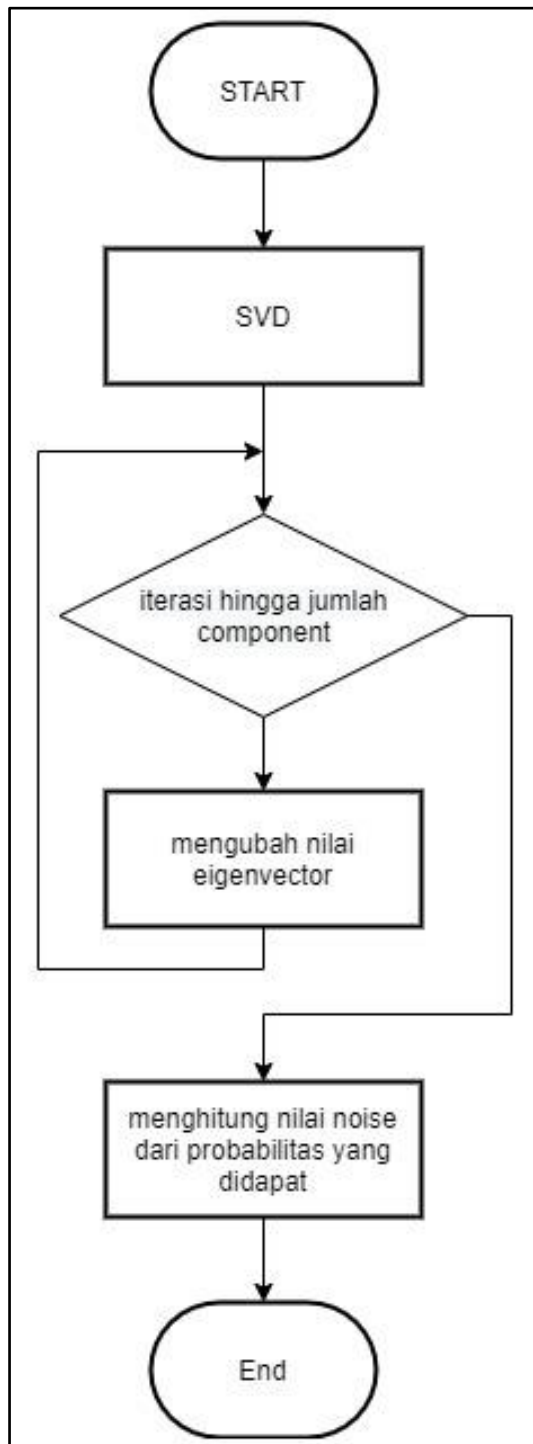


Gambar 3.1 Flowchart Sistem

Pada saat program mulai dijalankan, maka akan dimulai dengan memasukkan gambar yang dibutuhkan dengan menyamakan ukuran gambarnya. Dataset berupa input data yang merupakan gabungan dua gambar sisi kiri dan sisi kanan, dan label data yang merupakan identitas dari orang pada gambar tersebut disimpan. Selanjutnya dataset dibagi menjadi dataset latih dan dataset tes. Selanjutnya melakukan reduksi dimensi menggunakan *Principal Component Analysis* (PCA), sehingga ukuran data menjadi optimal dengan tetap mempertahankan *feature* penting. Setelah melakukan PCA, data pelatihan yang akan direduksi dipisah dengan data pelatihan yang akan digunakan untuk mendapat perbandingan hasil antara reduksi dimensi dengan yang tidak. Berikutnya adalah melatih data yang belum direduksi dengan model jaringan saraf tiruan, *Multi-Layer Perceptron* (MLP) yang dilatih menggunakan *backpropagation*. Selanjutnya, membuat model jaringan saraf tiruan baru dengan melatih data yang telah direduksi.

3.3 Flowchart PCA

Pada rancangan sistem yang akan digunakan, dijelaskan pada Gambar 3.1 bahwa dalam ada tahap untuk transformasi PCA digunakan untuk mereduksi dimensi sebuah data dengan menghilangkan data – data terikat dengan yang tersisa hanyalah data – data independen yang tidak dapat terikat. Penjelasan bagaimana proses sederhana PCA dapat mereduksi dimensi untuk dijelaskan pada Gambar 3.2.



Gambar 3.2 Flowchart PCA

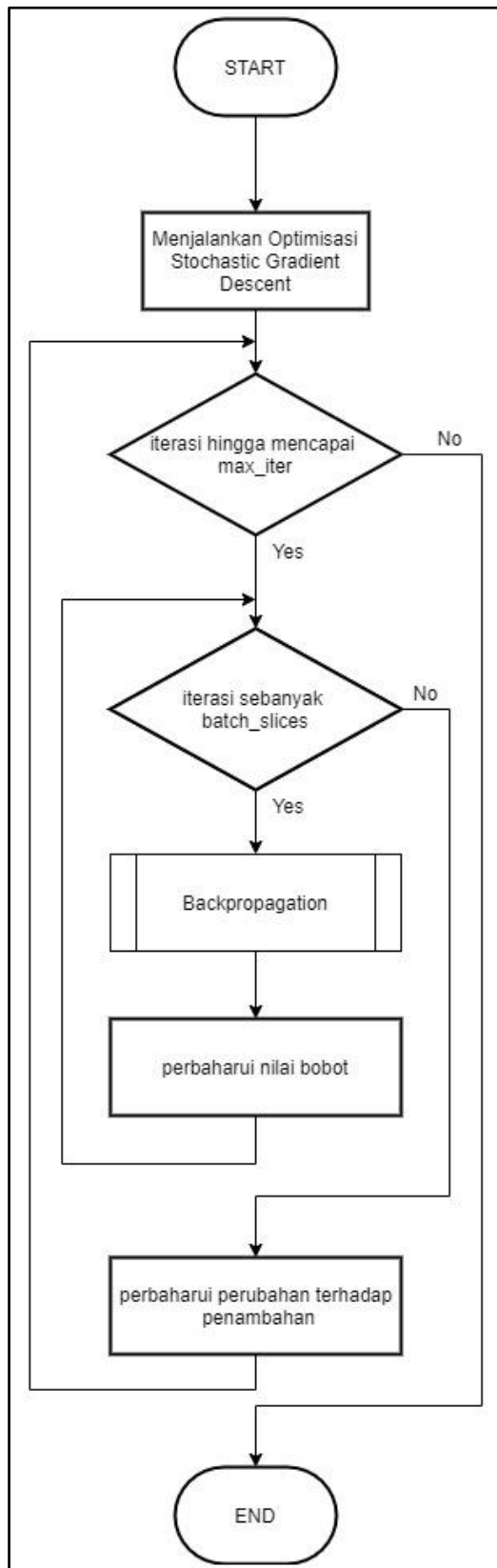
Ketika *class* PCA pada *library* python dijalankan, dilakukan beberapa inisiasi variabel guna berjalannya PCA. Setelah itu tahap ‘SVD’ atau ‘*Singular Value Decomposition*’ yaitu pemfaktoran matriks dengan mengurai suatu matriks ke dalam dua matriks uniter U dan V, dan sebuah matriks diagonal S yang berisi faktor skala yang disebut dengan nilai singular(Reza, 2020).

Setelah itu tahap membalikkan eigenvector dengan SVD pada nilai U dan nilai V. setelah itu mendapatkan nilai varian dari setiap data singular yang ada. Berikutnya adalah melakukan proses untuk bilangan dari komponen yang ada, dengan menentukan nilai bilangan dari komponen yang ada sesuai kebutuhan fitur. Berikutnya adalah tahap penentuan kovarian dari *noise* yang didapat secara probalistik dengan melihat nilai mean.

Setelah medapati kebutuhan nilai PCA yang didapat, implementasi model PCA yang telah disesuaikan dengan transformasi. Pada transformasi yang dilakukan adalah dengan melakukan *dot product* antara array yang pada data dengan array model PCA yang telah disesuaikan.

3.4 Implementasi Jaringan Saraf Tiruan

Pada implementasi jaringan saraf tiruan dilakukan menggunakan *library Multi-Layer Perceptron* (MLP) untuk dapat membuat model jaringan saraf tiruan. Pada Gambar 3.3 akan dijelaskan bagaimana *class* MLP melakukan penyesuaian model untuk dapat digunakan dan melatih data.



Gambar 3.3 Flowchart Multi-Layer Perceptron (MLP)

Pada awal melakukan model fitting untuk melatih model maka akan diperlukan ‘SGD’, yaitu *Stochastic Gradient Descent*, yang merupakan metode pengulangan untuk mengoptimalkan function objective dengan menggunakan kemungkinan paling cocok dengan dataset yang ditentukan secara acak dengan estimasi per subset dari data yang ada.

Setelah dilakukannya SGD, maka selanjutnya akan dilakukan iterasi, hingga jumlah iterasi yang diinginkan. Pada saat memulai maka akan menjalankan membagi pekerjaan menjadi beberapa kelompok atau disebut ‘*batch*’, dan akan menjalankan semua kelompok yang ada hingga keseluruhan data. Data pertama akan dilatih dengan backpropagation, setelah melakukan tahap backpropagation yang akan dijelaskan nanti, dilakukan proses akumulasi loss yang didapat, lalu melakukan pembaharuan nilai *weight*.

Setelah menyelesaikan satu bagian batch, maka akan didapatkan nilai *loss* dari iterasi batch sebelumnya dengan nilai akumulasi loss pada batch dibagi oleh ukuran atau besar data keseluruhan data pelatihan yang digunakan. Setelah itu memperbaharui perubahan yang ada, dan melakukan iterasi selanjutnya, untuk dapat menyelesaikan keseluruhan iterasi yang ada hingga akhir.

Setelah sebelumnya dijelaskan bahwa pada MLP melatih data menggunakan Backpropagation, Pada Gambar 3.4 dijelaskan bagaimana proses backpropagation yang akan terdapat dalam akan berlangsung untuk mengoptimalkan dan mendapatkan nilai loss dalam pelatihan data tersebut.



Gambar 3.4 Flowchart Backpropagation Pada MLP

Pada tahap backpropagation, diketika awal dimulai akan dilakukan tahap *forward propagate* (peramban maju), yaitu tahap untuk melakukan iterasi sebanyak jumlah *hidden layer* yang digunakan dengan menjalankan fungsi aktivasi yang digunakan, setelah itu akan mendapatkan nilai aktivasi. Setelah melalui tahap peramban maju selanjutnya melakukan mendapatkan nilai *loss* sementara setelah itu, menggunakan aturan regularisasi pada *loss*, baru melakukan *backward propagate* (peramban mundur). Dengan pertama mendapatkan nilai delta yang didapat dari aktivasi sebelumnya, dan melakukan penghitungan *loss gradient* pada layer terakhir. Kembali dengan melakukan iterasi terhadap hidden layer yang ada dengan mendapatkan nilai *loss*, gradien koefisien, dan gradien pemotongan yang nantinya nilai ini akan dikembalikan pada fungsi penyesuaian model pada MLP.