



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk menggubah, 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 PERANCANGAN SISTEM

#### 3.1 Metodologi Penelitian

Metode penelitian yang akan digunakan dalam penelitian ini adalah sebagai berikut.

##### 1. Studi Literatur

Melakukan kajian pustaka terhadap penelitian-penelitian ataupun *paper*, *ebook*, serta beberapa referensi pembelajaran lainnya.

##### 2. Analisis Kebutuhan

Melakukan analisis terhadap fitur-fitur yang dibutuhkan di dalam perancangan dan pembangunan aplikasi.

##### 3. Perancangan Aplikasi

Merancang aplikasi dengan membuat diagram untuk menunjukkan alur proses dari aplikasi yang akan dibuat dan melakukan perancangan terhadap tampilan antarmuka aplikasi.

##### 4. Pembangunan Aplikasi

Membangun aplikasi dengan menggunakan bahasa pemrograman C#.

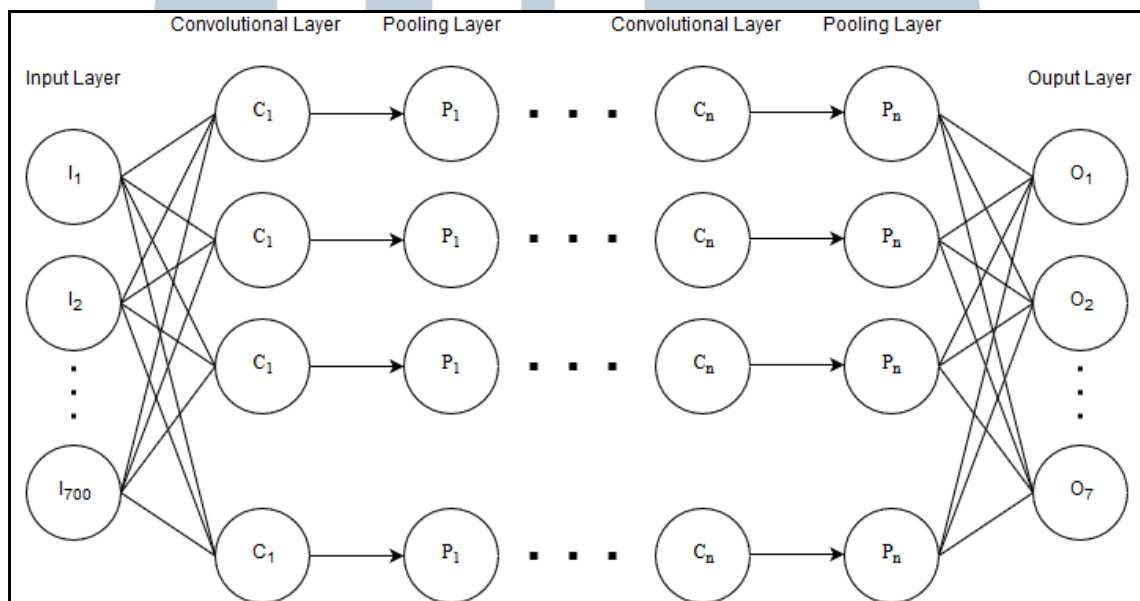
##### 5. Pengujian Aplikasi

Menguji aplikasi yang telah dibuat dengan menggunakan data hasil pindaian 3 dimensi dari wajah dengan menggunakan metode PCA (*Principal Component Analysis*) pada algoritma *convolutional neural networks*.

## 6. Penulisan laporan

Dilakukan pencatatan terhadap setiap langkah yang dilakukan dan informasi yang didapatkan selama proses penelitian berlangsung. Pencatatan dilakukan dengan cara penulisan laporan.

### 3.2 Arsitektur Convolutional Neural Network



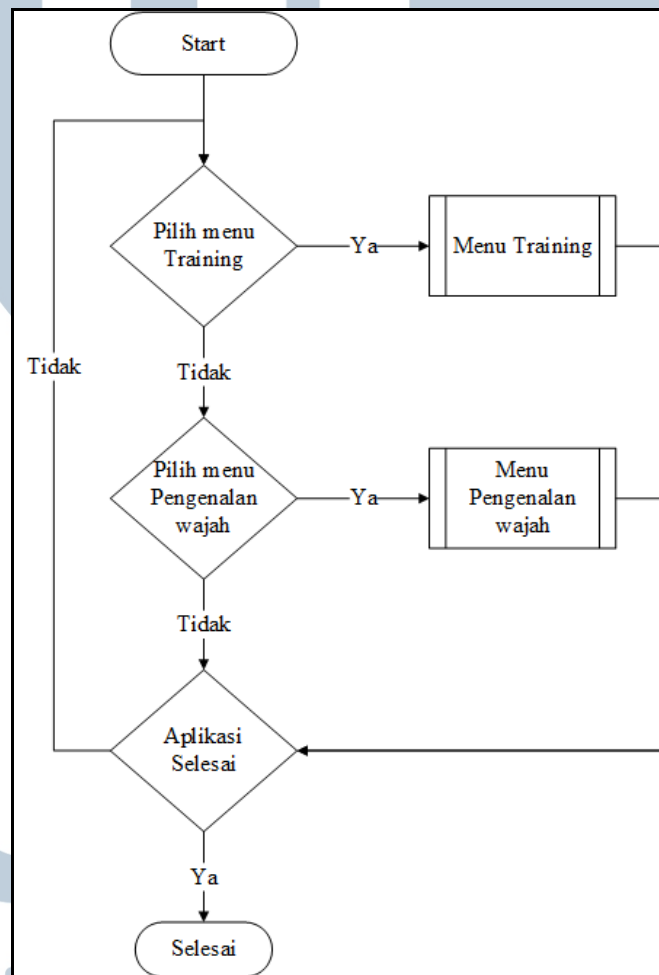
Gambar 3.1 Arsitektur Convolutional Neural Network

Pada Gambar 3.1 merupakan arsitektur *convolutional neural network* yang dipakai pada sistem. *Input layer* memiliki 700 *input node* yang berdasarkan 700 buah gambar wajah yang telah diproses menggunakan *library PCA (Principal Component Analysis)*. *Convolutional layer* merupakan bagian dari *hidden nodes layer* yang berisikan kumpulan *hidden nodes* yang berjumlah 1, 5, 10, 20, 50, 100, sesuai dengan jumlah *hidden nodes* yang dipakai pada penelitian ini. *Ouput layer* memiliki jumlah 7 *nodes* yang merupakan hasil dari *convolutional layer* dan *pooling layer*.

### 3.3 Perancangan Sistem

Perancangan sistem dilakukan agar dapat memahami alur kerja dari sistem tersebut secara keseluruhan.

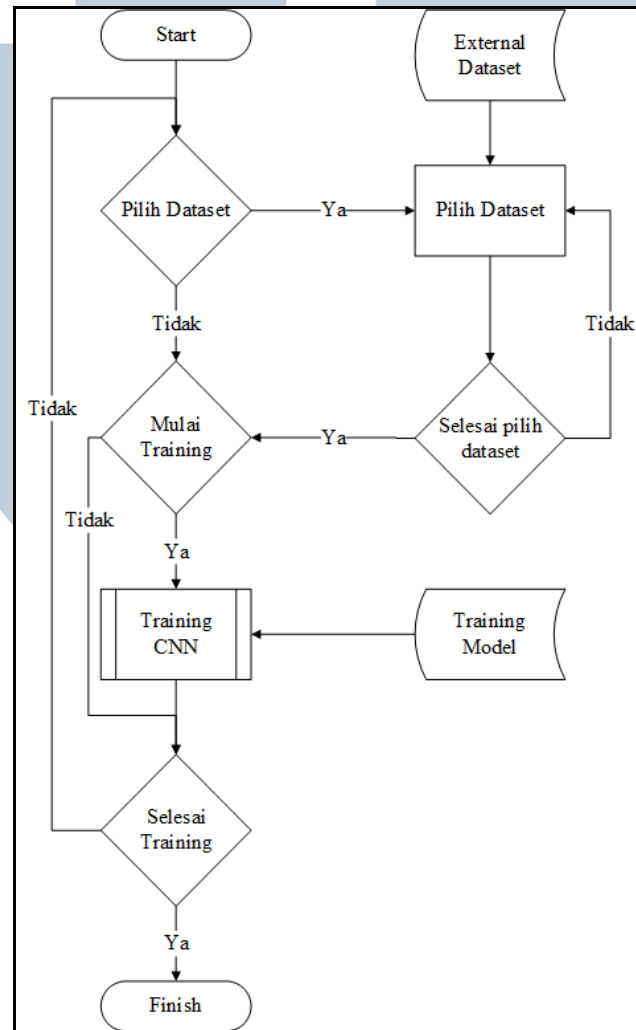
#### 3.3.1 Flowchart



Gambar 3.2 Flowchart Menu Utama

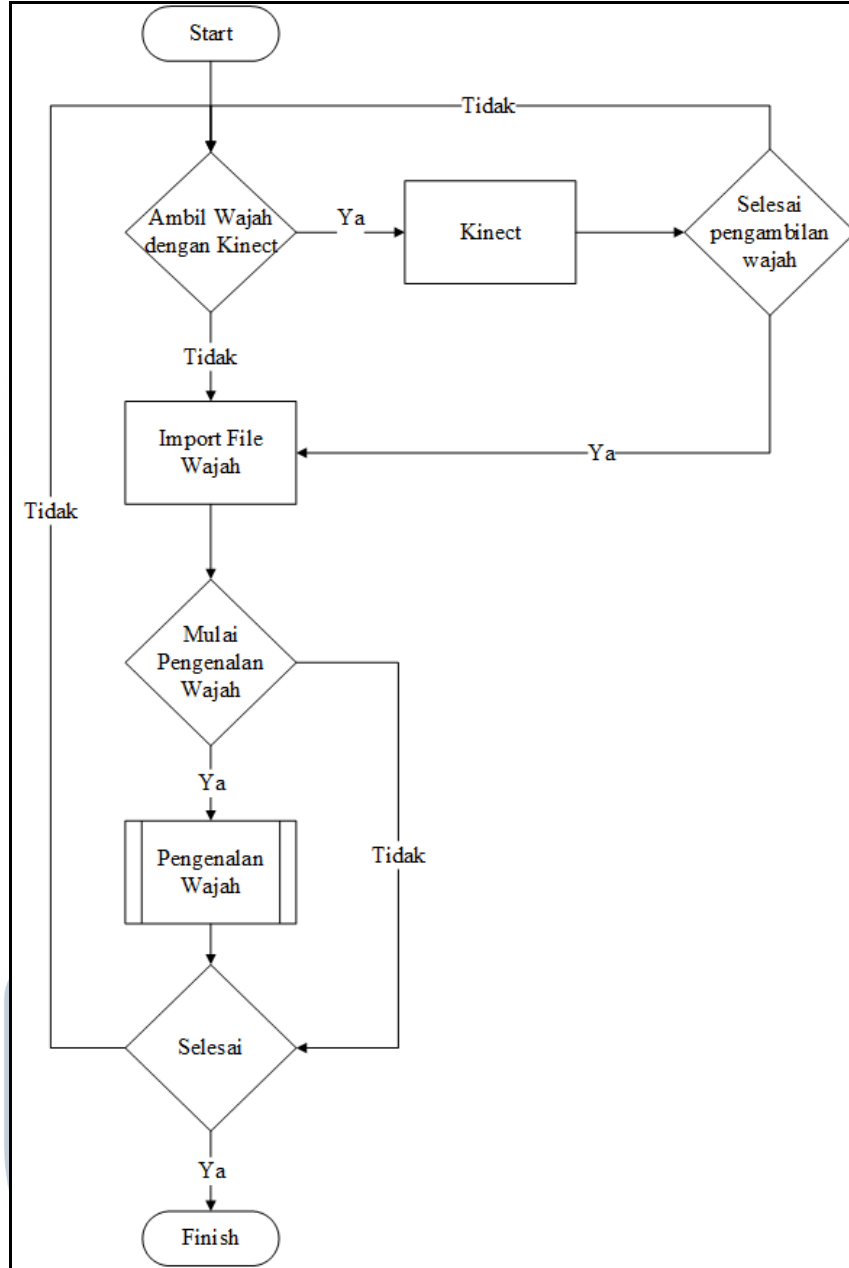
Pada Gambar 3.2 merupakan *flowchart* dari menu utama dari sistem. Pada menu ini terdapat dua pilihan menu, *training* dan pengenalan wajah. Menu *training* merupakan menu

untuk melakukan *training* pada *dataset*. Pada menu pengenalan wajah menu untuk melakukan prediksi wajah dari sebuah foto.



Gambar 3.3 Flowchart Menu Training

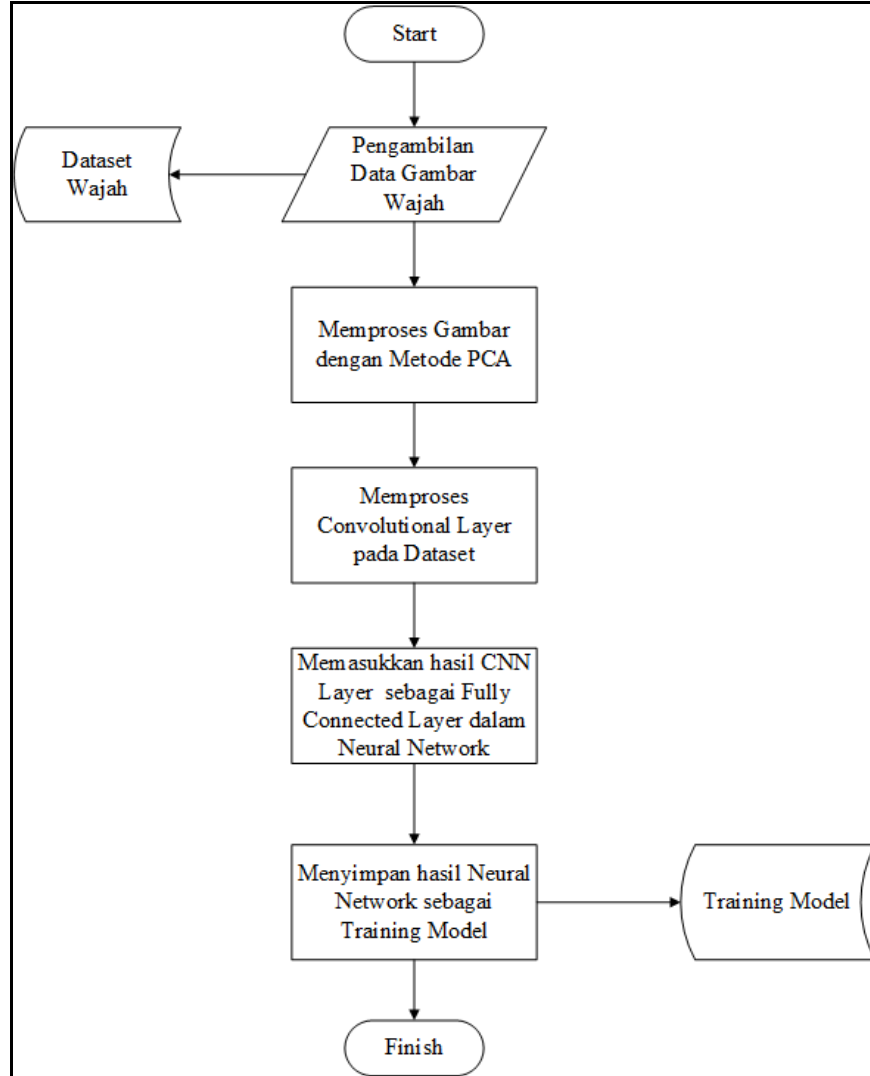
Gambar 3.3 merupakan *flowchart* dari halaman *training*, pada halaman ini terdapat dua pilihan yaitu pilih *dataset* yang dapat memasukan *dataset* baru untuk dilakukan *training* agar menjadi *training* model yang baru. Jika tidak memasukkan *dataset* yang baru *training* dilakukan menggunakan *training* model yang sudah ada.



Gambar 3.4 Flowchart Menu Pengenalan Wajah

Gambar 3.4 merupakan *flowchart* dari halaman pengenalan wajah. Halaman ini digunakan untuk melakukan prediksi wajah yang dapat dilakukan dengan langsung memfoto wajah dan dengan cara menggunakan *file* foto yang sudah ada.

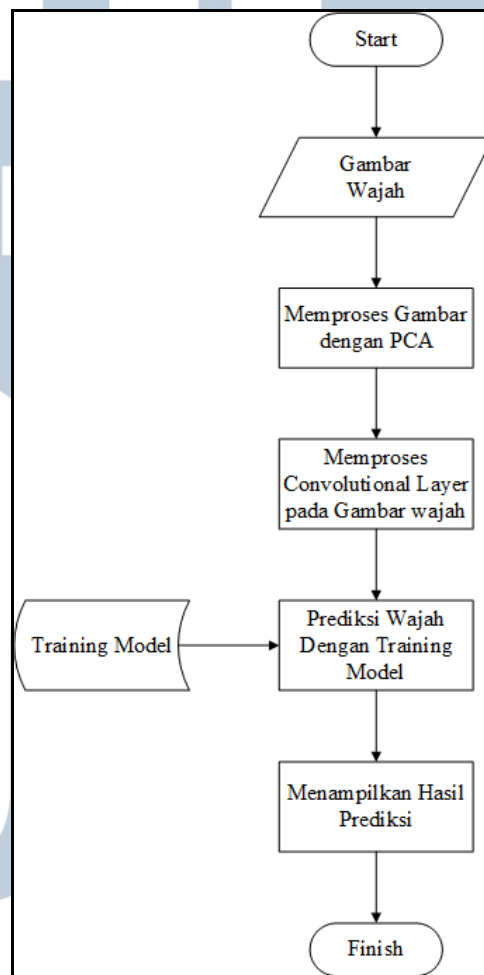
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.5 Flowchart Training CNN

Gambar 3.5 merupakan untuk melakukan *training* CNN menggunakan Matlab. *Training* dilakukan dengan mengambil data wajah dari foto atau gambar yang ada pada *dataset*, kemudian foto tersebut diproses dengan menggunakan metode PCA, setelah itu proses dilanjutkan pada *convolutional layer* untuk mendapatkan fitur bentuk wajah dan melakukan reduksi pada fitur-fitur lain sehingga membentuk satu matriks.

Kemudian memasukan hasil dari *convolutional layer* sebagai *fully connected layer* dalam bentuk *neural network* dengan mengambil matriks yang telah dibuat lalu dimasukan ke dalam *neural network* sebagai *weight*. Hasil dari *neural network* disimpan menjadi *file* Matlab yang berisi nilai dari matriks *weight* dan label untuk melakukan prediksi, kemudian disimpan menjadi *training model*.



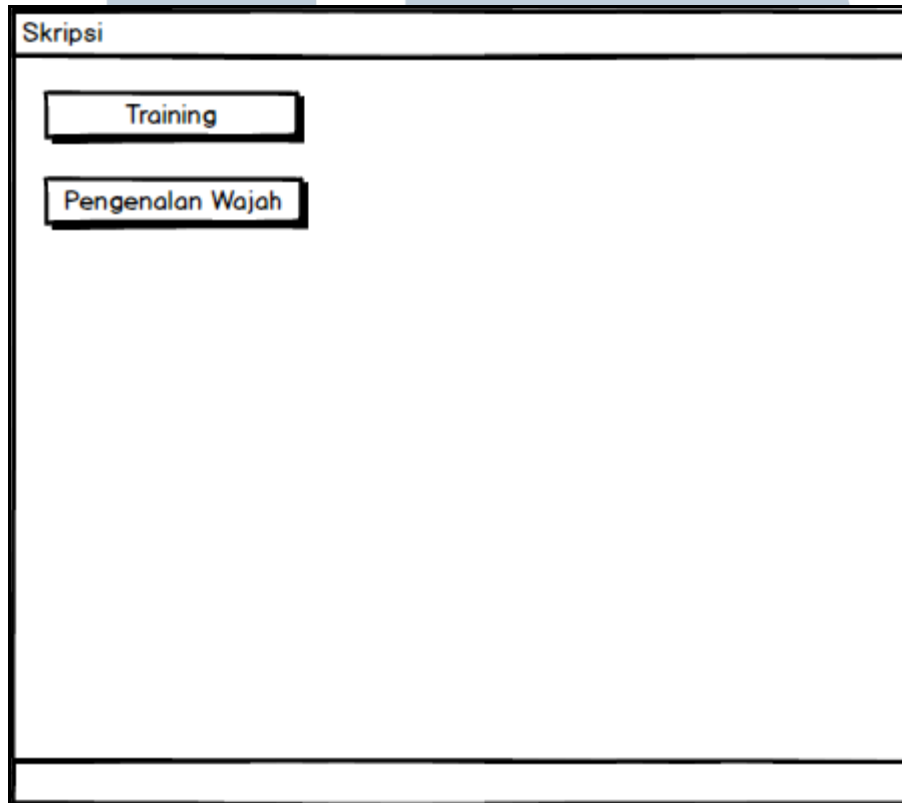
Gambar 3.5 Flowchart Pengenalan Wajah

Gambar 3.5 merupakan *flowchart* dari prediksi wajah dengan menggunakan Matlab. Foto wajah diproses dengan PCA untuk diambil fitur-fitur wajah, kemudian hasil matriks yang telah dibuat pada proses *training* dibandingkan dengan nilai matriks pada *training*



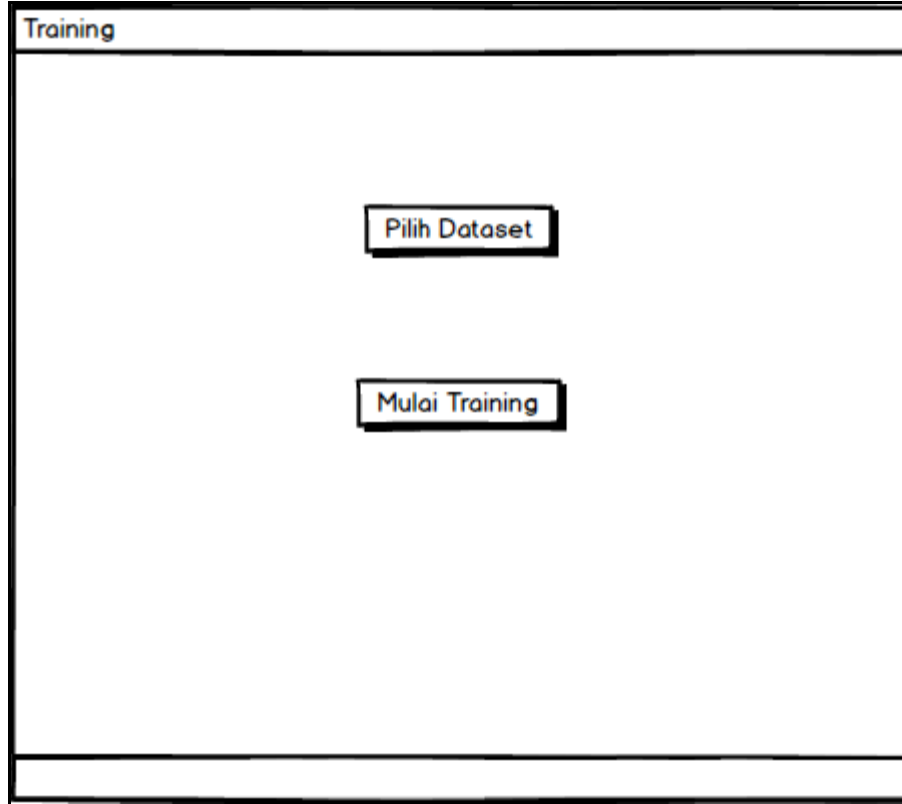
model untuk mendapatkan presentase kemiripan matriks gambar dengan matriks label. Hasil dari foto yang diprediksi ditampilkan dengan presentase setelah dicocokkan dengan *training* model.

### 3.3.2 Desain Antarmuka



Gambar 3.6 Desain Antarmuka Menu Utama

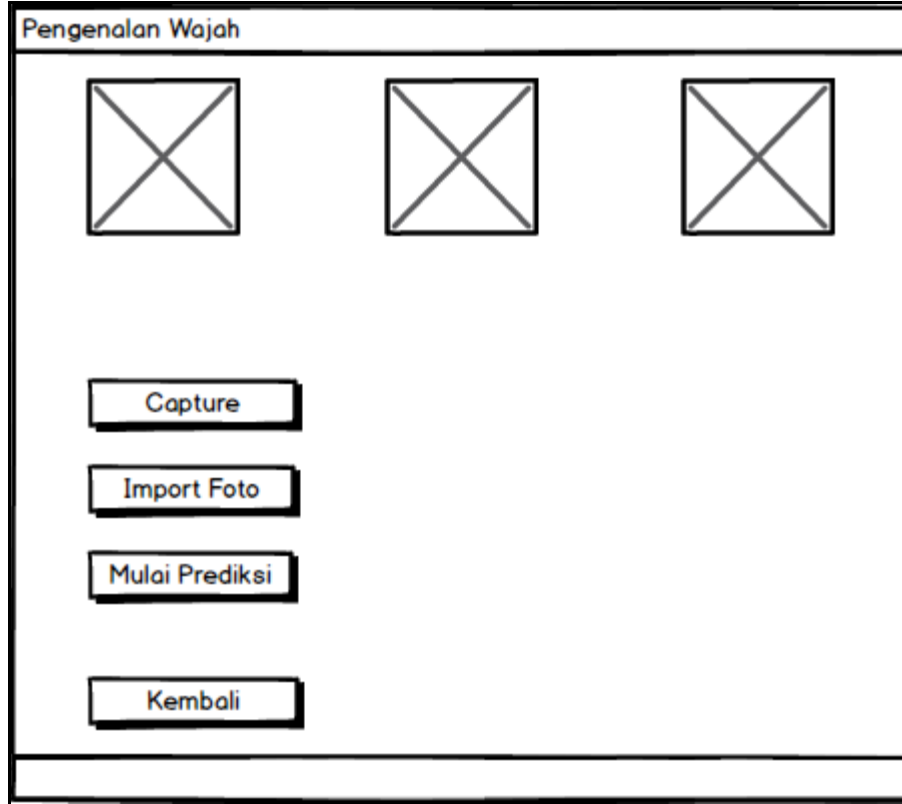
Gambar 3.6 merupakan desain antarmuka dari menu utama sistem ini, pada menu ini terdapat dua tombol untuk berpindah menu. Tombol *training* untuk berpindah pada menu *training*, halaman tersebut digunakan untuk melakukan *training* pada *dataset*. Tombol pengenalan wajah untuk masuk ke menu prediksi, yang digunakan untuk melakukan prediksi wajah pada sebuah foto/gambar.



Gambar 3.7 Desain Antarmuka Menu Training

Gambar 3.7 merupakan desain dari menu *training*, pada halaman ini terdapat dua tombol yaitu pilih *dataset* dan mulai *training*. Tombol pilih *dataset* digunakan untuk memilih *dataset* baru, jika sudah memilih *dataset*, tekan tombol mulai *training* untuk memulai *training*. Jika tidak memilih *dataset* baru dapat langsung memulai *training* dengan menggunakan *dataset* yang sudah ada.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



Gambar 3.8 Desain Antarmuka Menu Pengenalan Wajah

Gambar 3.8 merupakan desain dari menu pengenalan wajah. Pada menu ini terdapat tiga *picture box* yang digunakan untuk menampilkan gambar dari kamera Kinect. Tombol *capture* digunakan untuk mendapatkan foto hasil dari kamera Kinect. *Import* foto digunakan untuk memilih foto yang sudah ada untuk dilakukan prediksi. Tombol mulai prediksi digunakan untuk memulai pengenalan wajah pada foto yang sudah dipilih. Tombol kembali digunakan untuk kembali pada menu utama.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A