

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Metode deteksi citra pada tugas akhir ini mengacu pada beberapa hasil penelitian yang sudah dilakukan diantaranya adalah sebagai berikut.

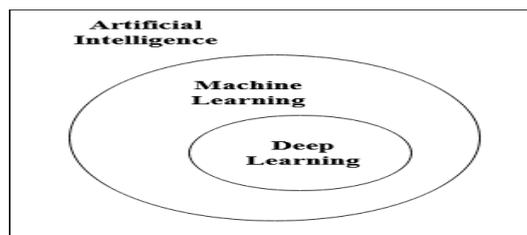
1. Nafis Irtija, Mahsius Sami dan Md Atiqur Rahman Ahad dari *University of Dhaka, Bangladesh*, melakukan penelitian dengan menggunakan metode *facial landmarks* untuk deteksi kantuk pada bagian mata. Pada penelitian tersebut menggunakan 606 gambar sebagai *dataset*, hasil penelitian menunjukkan tingkat akurasi sebesar 87,79 % [9].
2. Isha Gupta, Novesh Garg, Apoorva Aggarwal, Nitin Nepalia dan Bindu Verma dari *Jaypee Institute of Information Technology, India*, melakukan penelitian dengan menggunakan metoda *facial landmarks* untuk deteksi kantuk pada bagian mata dan bibir. Pada penelitian tersebut menggunakan 100 sampel sebagai uji coba dengan menghasilkan tingkat akurasi deteksi sebesar 90% [8].
3. Akalya Chellappa, Mandi Sushmanth Reddy, R.Ezhilarasie, S.Kanimozhi Suguna dan A. Umamakeswari dari *School of Computing, SASTRA Deemed to be University, Thanjavur*, melakukan penelitian dengan menggunakan metode *facial landmarks* untuk deteksi kantuk dengan mata sebagai acuan deteksi. Penelitian dilakukan dengan memperhatikan intensitas cahaya dan deteksi dilakukan setiap mata kali berkedip selama 0.5 detik. Hasil penelitian memiliki tingkat akurasi yang tinggi dengan melakukan sepuluh kali percobaan [10].
4. Yulia Gizatdinova dan Veikko Surakka melakukan penelitian dengan menggunakan metode *facial landmarks* untuk mendeteksi ekspresi wajah. Pada penelitian tersebut menggunakan mata, hidung dan mulut sebagai acuan dengan beberapa deteksi ekspresi wajah seperti senang, sedih, takut, marah, kaget dan menjijikkan. Hasil penelitian tersebut menghasilkan tingkat akurasi deteksi sebesar 90% [11].

5. Jilliam Maria Diaz Barros, Bruno Mirbach, Frederic Garcia dan Kiran Varanisa, Didier Stricker dari *Technical University of Kaiserslautern, Germany*, melakukan penelitian dengan menggunakan metode *facial landmarks* dan menggabungkan metode filter kalman untuk deteksi posisi kepala secara *real time*. Hasil deteksi pada penelitian tersebut menghasilkan kecepatan proses deteksi 40 *frame rate per second* (FPS) dimana hasil yang didapatkan sama ketika menggunakan *Head Pose Estimation* (HPE) *in fused framework* [12].

Perancangan sistem pada tugas akhir menggunakan dua parameter deteksi yaitu ketika mata tertutup beberapa detik dan menguap, selain itu pengujian sistem dilakukan dengan menguji tingkat akurasi deteksi, pengujian berdasarkan intensitas cahaya dan pengujian berdasarkan jarak deteksi.

2.2 Artificial Intelligence (AI)/ Kecerdasan Buatan

Artificial Intelligence (AI) merupakan bidang ilmu yang mempelajari bagaimana membuat mesin dapat memiliki kecerdasan buatan seperti manusia. AI merupakan sebuah sistem cerdas yang ditanamkan pada mesin dimana sistem cerdas dapat didefinisikan sebagai pengetahuan dan penalaran yang dapat dilakukan oleh mesin untuk memecahkan suatu masalah contohnya, mengenali objek, pengambilan keputusan, klasifikasi data, pengenalan wajah dan lain-lain [4]. Ada berbagai bidang yang dapat diselesaikan dengan menggunakan teknologi AI diantaranya pada bidang *ecommerce*, industri, transportasi, kesehatan, pendidikan, ritel dan bidang-bidang lainnya. Secara garis besar AI merupakan bagian dari *machine learning* dan *deep learning* dimana AI menjadi landasan atau akar terhadap bidang ilmu yang berkembang hingga saat ini. Pada dasarnya AI merupakan induk dari *machine learning* dan *deep learning* seperti terlihat pada Gambar 2.1 [13].

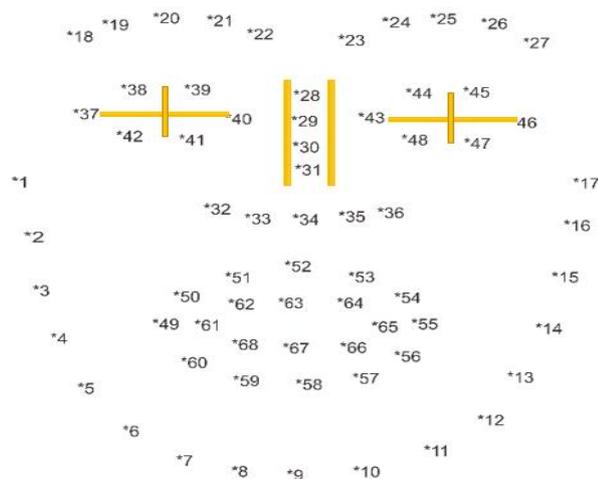


Gambar 2.1. Diagram venn *Artificial Intelligence* (AI) [13]

2.3 Face Recognition/ Pengenalan Wajah

Face recognition merupakan salah satu teknik dalam *computer vision* yang dapat digunakan untuk mendeteksi citra wajah. Pada dasarnya komputer tidak seperti manusia yang dapat langsung mengenali seperti manusia, melainkan komputer mengenali citra berdasarkan nilai piksel. Citra dapat didefinisikan ke dalam fungsi $f(x,y)$ dua dimensi dan $f(x,y,z)$ untuk tiga dimensi dimana f merupakan intensitas, x dan y adalah koordinat spasial dan z adalah kedalaman citra. Citra dikenali sebuah komputer dalam bentuk matriks terdiri dari baris dan kolom yang menyatakan elemen matriks untuk merepresentasikan tingkat keabuan pada citra. Nilai intensitas pada titik x dan y memiliki skala dari 0 sampai 255 atau $[0,255]$, dalam hal ini nilai intensitas 0 menyatakan hitam dan nilai intensitas 255 menyatakan putih [13].

Salah satu teknik pengenalan wajah yang populer digunakan adalah teknik *active appearance models* (AAMs) dari Cootes, Edwards dan Taylor. Cara kerja teknik ini dalam pengenalan wajah yaitu dengan menemukan titik koordinat dan melakukan analisis pengenalan wajah atas dasar posisi atau jarak relatif pada bagian wajah yang khas seperti bibir, hidung dan mata. Teknik AAMs dimodelkan dalam bentuk citra yang dikodekan ke dalam bentuk titik koordinat bagian wajah seperti pada Gambar 2.1, metode yang sudah dikembangkan dengan menggunakan teknik AAMs salah satunya adalah metode *facial landmarks detection* [14].



Gambar 2.2. Titik koordinat *facial landmarks* [14]

2.4 Open Source Computer Vision Library

Open Source Computer Vision Library (OpenCV) merupakan pustaka perangkat lunak terdiri dari lebih 2.500 algoritma yang mencakup pembelajaran mesin mengenai pengolahan citra digital. OpenCV secara resmi dirilis pada tahun 1999 oleh tim penelitian yang dipimpin Gary Bradsky. Pustaka OpenCV yang bersifat *open source* sehingga banyak peneliti yang telah mengembangkan berbagai algoritma khususnya algoritma deteksi wajah [15]. Pustaka ini dapat tulis dalam bahasa pemrograman C/C++, Java dan python dan dapat berjalan pada perangkat lunak Linux, Windows, dan Mac OS. Dalam pustaka OpenCV terdapat pustaka yang dapat digunakan untuk mendukung metode *facial landmarks detection* diantaranya adalah sebagai berikut [16].

1. *Cv2*

Pustaka *cv2* dapat digunakan untuk menyimpan citra, membaca citra, menulis citra dan lain-lain.

2. *Dlib*

Pustaka *dlib* digunakan untuk memperkirakan titik koordinat pada detector *landmarks* yang memetakan struktur wajah.

3. *Imutils*

Pustaka *Imutils* digunakan untuk mengubah ukuran citra.

4. *Scipy.spatial*

Pustaka *Scipy.spatial* digunakan untuk mengubah titik *facial landmarks* menjadi vektor sehingga dapat dilakukan perhitungan lebar mata maupun bibir.

5. *General Purpose Input/Output (GPIO)*

Pustaka GPIO digunakan untuk mengaktifkan pin pada Raspberry PI 4 Model B.

6. *Time*

Pustaka *time* digunakan untuk mengaktifkan fitur *delay* pada Raspberry PI

4 model B.

Penggunaan pustaka tersebut perlu digunakan untuk mendukung sistem pendeteksi kantuk agar sistem dapat bekerja dengan baik, penggunaan pustaka tersebut dapat ditunjukkan pada Gambar 2.3.

```
1 from scipy.spatial import distance
2 from imutils import face_utils
3 import imutils
4 import dlib
5 import cv2
6 import RPi.GPIO as GPIO
7 import time
```

Gambar 2.3. Implementasi program dengan menggunakan bahasa python untuk pustaka yang digunakan pada sistem

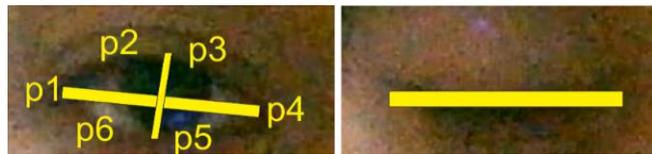
2.5 Facial Landmarks Detection

Metode *facial landmarks* digunakan untuk mendeteksi bagian wajah termasuk pada bagian mata dan bibir yang digunakan untuk objek deteksi kantuk. Metode ini memberikan titik koordinat bagian wajah terdiri dari 68 titik *landmarks*, titik tersebut termasuk hidung, mata dan bibir seperti terlihat pada Gambar 2.2. Untuk mendeteksi perubahan mata dan bibir dihitung berdasarkan titik pusat dimana titik pusat memiliki koordinat sebanyak 68 koordinat [9]. Berdasarkan koordinat ini dapat dideteksi posisi mata pada koordinat pusat 49 dan 55 serta untuk bibir dapat dideteksi pada koordinat 49 dan 55. Dalam memprediksi rasa kantuk dapat dihitung dengan menggunakan aspek rasio mata dan bibir, dengan mengukur jarak antara mata dan bibir seperti yang ditunjukkan pada Gambar 2.4 dan Gambar 2.5. Penentuan apakah pengemudi kendaraan berat mengantuk didasarkan ketika mata tertutup lebih lama atau pada saat menguap maka hal ini dapat disimpulkan bahwa pengemudi terindikasi mengantuk. Persamaan yang digunakan untuk menentukan nilai rasio mata dan mulut dapat dilihat pada Persamaan 2.1 untuk rasio mata dan Persamaan 2.2 untuk rasio bibir [8].

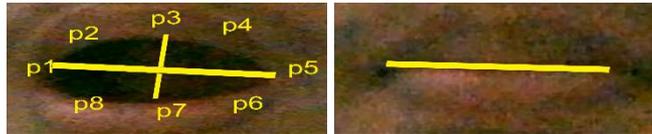
$$\text{AspekRasioMata} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2 \times \|p_1 - p_4\|} \quad (2.1)$$

$$\text{AspekRasioBibir} = \frac{\|p_2 - p_8\| + \|p_4 - p_6\|}{2 \times \|p_1 - p_5\|} \quad (2.2)$$

dimana p_1, p_2, \dots, p_8 merupakan nilai koordinat $p_n(x_i, y_j)$ dari *facial landmarks* mata dan bibir seperti yang ditunjukkan pada Gambar 2.1. Dari Persamaan 2.1 dan Persamaan 2.2 dapat dihitung nilai aspek rasio dengan menggunakan rumus *euclidean* dimana rumus ini menghitung jarak objek berdasarkan koordinat seperti pada Gambar 2.6 dan Persamaan 2.3.

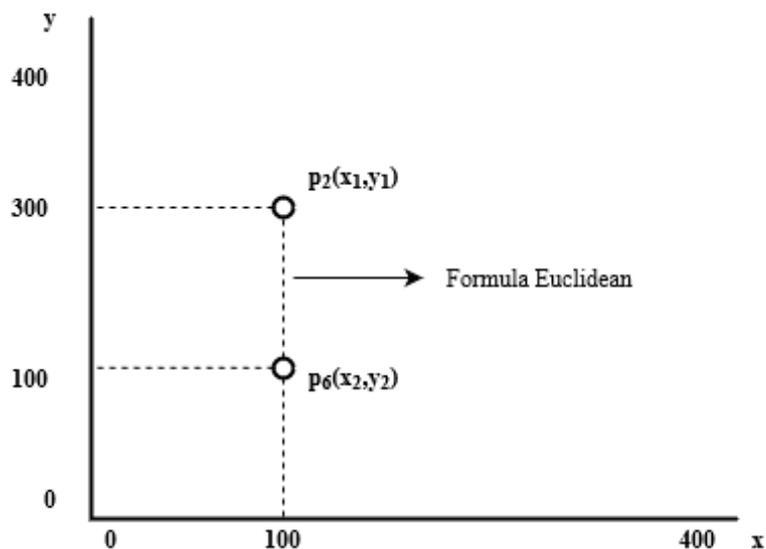


Gambar 2. 4. Rasio mata tertutup dan terbuka dengan *landmarks detection* [8]



Gambar 2. 5. Rasio bibir tertutup dan terbuka dengan *facial landmarks detection* [8]

$$\text{JarakEuclidean} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (2.3)$$



Gambar 2.6 Konsep perhitungan nilai rasio dengan menggunakan rumus *Euclidean*