# DAFTAR PUSTAKA

[1]   A. R. Gifari Alim Prakasa, "Prototype Sistem Kunci Pintu Berbasis," 2017.

[2]   V. Raja, "Pengenalan Wajah Sebagai Sistem Kendali Buka Tutup Pintu Ruangan Dengan Metode Support Vector Machine," 2018.

[3]   E. H. H. guntoro Yoyo Somantri, "Rancang Bangun Magnetic Door Lock Menggunakan Keypad Dan Solenoid Berbasis Mikrokontroler Arduino Uno," *Electrans*, vol. 12, no. 1, pp. 39–48, 2013.

[4]   M. Miftah, J. T. Informatika, U. Dian, N. Semarang, P. Wajah, and T. Face, "Pengamanan Laptop Menggunakan Pengenalan Wajah Berbasis Triangle Face," *J. Appl. Intell. Syst. UDINUS*, vol. 1, no. 1, pp. 22–35, 2016.

[5]   S. Dhian, "Analisa Algoritma Sistem Keamanan Poin Minutiae Pada Hp Compact 2210B Notebook Pc," *J. Univ. Gunadarma*, vol. 6, no. 01, pp. 18–21, 2012.

[6]   B. Septian, A. Wijayanto, F. Utaminingrum, and I. Arwani, "Face Recognition Untuk Sistem Pengaman Rumah Menggunakan Metode HOG dan KNN Berbasis Embedded," *Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 3, pp. 2774–2781, 2019.

[7]   F. Muhammad, "Sistem Keamanan Akses Pintu Masuk Menggunakan Face Recogition Berbasisi Raspberry Pi 3," 2018.

[8]   Liton Chandra Paul and Abdulla Al Sumam, "Face Recognition Using Principal Component Analysis Method," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 1, no. 9, pp. 2278–1323, 2012.

[9]   M. Mariappan, "A LabVIEW Design for Frontal and Non- Frontal Human Face Detection System in Complex Background," *Appl. Mech. Mater*, vol. 490–491, no. 1, pp. 1259–1266, 2014.

[10]   M. Çar and F. Özen, "A Face Recognition System Based on Eigenfaces Method," *Procedia Technol*, vol. 1, no. 1, pp. 118–123, 2012.

[11]   M. D. Malkauthekar, "Analysis of euclidean distance and manhattan distance measure in face recognition," *Third Int. Conf. Comput. Intell. Inf. Technol*, vol. 1, no. 4, pp. 3–7, 2013.

[12]   A. Dagar, M. T. E. C. E. Hod, C. Engineering, and F. Nagar, "Face Recognition Using Mahalanobis Distance In MATLAB," *Int. J. Technol. Res. Eng.*, vol. 5, no. 8, pp. 3499–3503, 2018.

[13]   Z. Fan, M. Ni, M. Sheng, Z. Wu, and B. Xu, "Principal component analysis integrating mahalanobis distance for face recognition," *Proc. - 2013 2nd Int. Conf. Robot. Vis. Signal Process. RVSP 2013*, pp. 89–92, 2013.

[14]   A. K. & A. Susanto, *Teori dan Aplikasi Pengolahan Citra*. CV. ANDI OFFSET, 2013.

[15]   T. Mulyono, K. Adi, and R. Gernowo, "Sistem Pengenalan Wajah dengan Metode Eigenface dan Jaringan Syaraf Tiruan (JST)," *Berk. Fis.*, vol. 15, no. 1, pp. 15–20, 2012.

[16]   M. üg. Çarıkçı and F. Özen, "A Face Recognition System Based on Eigenfaces Method," *Procedia Technol.*, vol. 1, pp. 118–123, 2012.

[17]   S. Bayu, A. Hendriawan, and R. Susetyoko, "Penerapan Face Recognition Dengan Metode Eigenface Dalam Intelligent Home Security," *Tugas Akhir Mhs. …*, pp. 1–7, 2009.

[18]   B. Utomo, "Matematika Eigenface Menggunakan Metrik Euclidean," in *Seminar Nasional Matematika dan Pendidikan Matematika2*, 2011, vol. 6, no. 3, pp. 44–52.

[19]   A. H. Mulyadi, L. Talangko, and A. Lawi, "Pengenalan Citra Wajah Menggunakan Metode Analisis Komponen Utama Dan Jarak Euclidean," *Progr. Stud. Stat. FMIPA Univ. Hasanuddin*, vol. 9, no. 6, pp. 1–14, 2014.
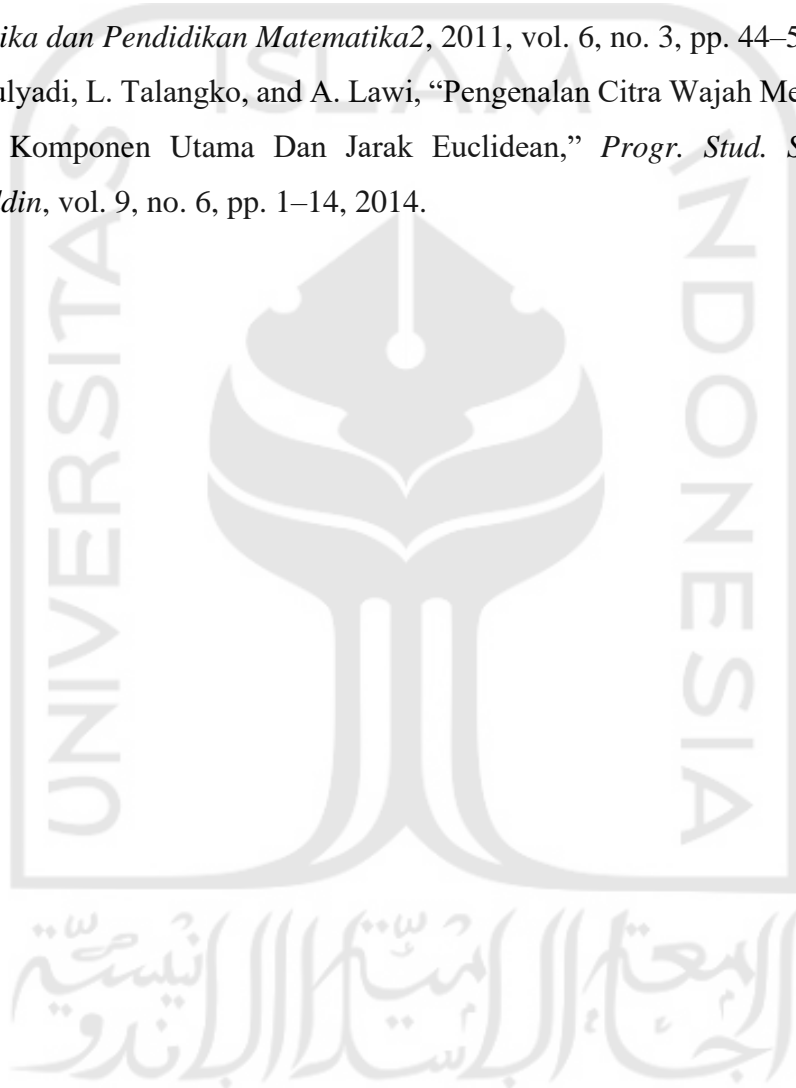
LAMPIRAN

Mathscript State **Operasi Citra** *Dataset*

```
Average_Face_Vector = mean(Normalisasi_Citra_Dataset,2);
A = [];
for i = 1 : JumlahDataset
phi =  double(Normalisasi_Citra_Dataset(:,i))-Average_Face_Vector;
A      = [ A phi ]; %Matrix A
end
L = A'*A;
[eigenvector_matrix_L,eigenvalue_matrix_L] = eig(L);
Vector_Eigen = [];
for i = 1 : size(eigenvector_matrix_L,2)
Vector_Eigen = [Vector_Eigen eigenvector_matrix_L(:,i)];
end
end
Matrix_Eigenface = A*Vector_Eigen;
Proyeksi_Dataset = [];
for i = 1 : size(Matrix_Eigenface,2)
Matrix_Bobot = Matrix_Eigenface' * A(:,i);
Proyeksi_Dataset = [Proyeksi_Dataset Matrix_Bobot];
End
```

Mathscript State **Pemerosesan Citra** *Dataset*

```
[Baris Kolom] = size(Wajah_Detek);
reshape_Wajahdetek = reshape(Wajah_Detek',Baris*Kolom,1);
temp = double(reshape_Wajahdetek)-AverageX;
Proyeksi_Datatest = Matrix_Eigenface'*temp;
Jarak_Euclidean = [ ];
for i=1 : size(Matrix_Eigenface,2)
  Math_Jarak_Euclidean = (norm(Proyeksi_Datatest-Proyeksi_Dataset(:,i)))^2;
  Jarak_Euclidean = [Jarak_Euclidean Math_Jarak_Euclidean];
End
[Baris_Jarak_Euclidean Kolom_Jarak_Euclidean] = size(Jarak_Euclidean);
reshape_Jarak_Euclidean = reshape(Jarak_Euclidean',Baris_Jarak_Euclidean*Kolom
[Jarak_Euclidean_Minimal Index_Jarak_Euclidean_Minimal] = min(Jarak_Euclidean);
```