

LAMPIRAN



SURAT KEPUTUSAN
REKTOR IIB DARMAJAYA
NOMOR : SK. 0604/DMJ/DFIK/BAAK/XI-22
Tentang
Dosen Pembimbing Skripsi
Semester Ganjil TA.2022/2023
Program Studi S1 Teknik Informatika

REKTOR IIB DARMAJAYA

- Memperhatikan :** 1. Bahwa dalam rangka usaha peningkatan mutu dan peranan IIB Darmajaya dalam melaksanakan Pendidikan Nasional perlu ditingkatkan kemampuan mahasiswa dalam **Skripsi**.
- Menimbang :** 1. Bahwa untuk mengefektifkan tenaga pengajar dalam Skripsi mahasiswa perlu ditetapkan **Dosen Pembimbing Skripsi**.
 2. Bahwa untuk maksud tersebut dipandang perlu menerbitkan Surat Keputusan Rektor.
- Mengingat :** 1. UU No.20 Tahun 2003 Tentang Sistem Pendidikan Nasional.
 2. Peraturan Pemerintah No.60 Tahun 2010 tentang Pendidikan Sekolah Tinggi
 6. Surat Keputusan Menteri Pendidikan Nasional Republik Indonesia No.165/D/0/2008 tertanggal 20 Agustus 2008 tentang Perubahan Status STMIK-STIE Darmajaya menjadi Informatics and Business Institute (IBI) Darmajaya
 7. STATUTA IBI Darmajaya
 8. Surat Ketua Yayasan Pendidikan Alfian Husin No. IM.003/YP-AH/X-08 tentang Persetujuan Perubahan Struktur Organisasi
 6. Surat Keputusan Rektor 0383/DMJ/REK/X-08 tentang Struktur Organisasi.
- Menetapkan**
- Pertama :** Mengangkat nama-nama seperti tersebut dalam lampiran Surat Keputusan ini sebagai Dosen Pembimbing Skripsi mahasiswa Program Studi S1 Teknik Informatika.
- Kedua :** Pembimbing Skripsi berkewajiban melaksanakan tugasnya sesuai dengan jadwal yang telah ditetapkan.
- Ketiga :** Pembimbing Skripsi yang ditunjuk akan diberikan honorarium yang besarnya sesuai dengan ketentuan peraturan dan norma penggajian dan honorarium IBI Darmajaya.
- Keempat :** Surat Keputusan ini berlaku sejak tanggal ditetapkan dan apabila dikemudian hari terdapat kekeliruan dalam keputusan ini, maka keputusan ini akan ditinjau kembali.

Ditetapkan di : Bandar Lampung
 Pada tanggal : 07 November 2022
 a.n. Rektor IIB Darmajaya,
 Dekan Fakultas Ilmu Komputer


 Dr. Sutedi, S.Kom., M.T.
 NIK. 00590203

1. Kepala Program Studi S1 Teknik Informatika
2. Yang bersangkutan
3. Arsip

Lampiran : Surat Keputusan Rektor IIB Darmajaya
 Nomor : SK.0604/DWI/DFIK/BAAY/XI-22
 Tanggal : 07 November 2022
 Perihal : Pembimbing Penulisan Skripsi Semester Ganjil TA. 2022/2023
 Program Studi Strata Satu (S1) Teknik Informatika

Judul Skripsi Dan Dosen Pembimbing Skripsi Semester Ganjil TA. 2022/2023
 Program Studi Strata Satu (S1) Teknik Informatika

No	NAMA	NPM	JUDUL	PEMBIMBING
37	Muhammad Rizky	1711010092	Multimedia Interaktif pada sistem layanan studio foto berbasis Android	Sepilla Arifa, S.Kom., MTI
38	Sabella Nur Afri	1811010138	Implementasi Fuzzy Topsis Pada Lomba Pimnapres Iib Darmajaya Berbasis Website	Sepilla Arifa, S.Kom., MTI
39	Ayeng Winanda	1911010034	Penerapan Algoritma Knuth Morris Pratt Terhadap Kelompok Bidang Keilmuan Mahasiswa Program Studi Teknik Informatika	Sepilla Arifa, S.Kom., MTI
40	Nanda Wahyu Prasetya	1911010033	E-Modul Pembelajaran Mata Kuliah Peminatan Kecerdasan Buatan Program Studi Teknik Informatika Iib Darmajaya Berbasis Android	Siti Nurhalia, S.Kom., MTI
41	Fadly Dwi Kurniawan	1911010039	Implementasi Teknologi Geolocation Pada Aplikasi Presensi Karyawan Iib Darmajaya	Siti Nurhalia, S.Kom., MTI
42	Mella Puspla A	1911010088	E-Modul Pembelajaran Mata Kuliah Peminatan Kecerdasan Buatan Program Studi Teknik Informatika Iib Darmajaya Berbasis Android	Siti Nurhalia, S.Kom., MTI
43	Andra Ramadan Pratama	1911010125	Implementasi Algoritma Winnowing Untuk Deteksi Tingkat Kemiripan Judul Skripsi berbasis Mobile	Sudi Mutiara, S.Kom., MTI
44	Dikki Datri Murdoko	1911010054	Face recognition Presensi Mahasiswa Iib Darmajaya Menggunakan Metode Eigenface	Sulyono, S.Kom., MTI
45	Fredy Fearyansyah	1711010094	E-Monitoring Skripsi Mahasiswa Dengan Menggunakan Algoritma CA.5 Berbasis WEB	Sulyono, S.Kom., MTI
46	WINDA OKTAVIA	1911010134	Monitoring aktivitas akademik mahasiswa program studi teknik informatika dengan menggunakan metode simple emit atribut rating technique (smart)	Sulyono, S.Kom., MTI
47	Helina Aqriyani	1911010154	Penerapan Metode Content Based Filtering Untuk Menentukan Top N Rekomendasi Riset Pada Pusat Unggulan (puj) Program Studi Teknik Informatika Berbasis Website	Suliyono, S.Kom., MTI
48	I Wayan Susila Agus Dharna	1911010066	Aplikasi Media Pembelajaran Pengenalan 3d Modeling & Animation Di Smtk Multimedia Menggunakan Metode Interactive Multimedia berbasis Android	Triwaili Rosandi, S.Kom., MTI
49	Rehan Reski R	1911010123	Perangkat Lunak Sistem Perkuliahan Praktisi dan Dosen Tamu Program Studi Teknik Informatika Iib Darmajaya Berbasis Web	Triwaili Rosandi, S.Kom., MTI
50	I Kadet Krisna	1911010069	E-Checklist Persyaratan sedang skripsi mahasiswa program studi TI Iib Darmajaya Berbasis Web	Triwaili Rosandi, S.Kom., MTI
51	Lugmanul Hakim	1911010059	Rancang Bangun Pengenalan Profile Provinsi Pada Pulau Sumatera Sebagai Media Pembelajaran Siswa Sekolah Dasar Menggunakan Teknologi 3d Augmented Reality	Triwaili Rosandi, S.Kom., MTI
52	Muhammad Hasip	1911010077	Sistem Cerdas E-Survey Dosen Pengampan Program Studi Teknik Informatika Menggunakan Fuzzy Inference System (FIS)	Yulmaini, S.Kom., M.Kom
53	Anita Dewi Purwati	1911010119	Sistem Cerdas Rekomendasi Peningkatan Lulusan Prodi Teknik Informatika Berdasarkan Kuisiomer Kepuasan Stakeholders Menggunakan Algoritma K-Means	Yulmaini, S.Kom., M.Kom
54	Siti Uari	1911010072	Sistem Cerdas Untuk Merekomendasikan Metode Pembelajaran Berdasarkan E-Survei Alumni Program Studi Teknik Informatika Menggunakan Algoritma Naive Bayes	Yulmaini, S.Kom., M.Kom



Institut Informatika & Bisnis

DARMAJAYA

Yayasan Alfian Husin

Jl. Zainal Abidin Pagar Alam No. 93 Bandar Lampung 35142 Telp 787214 Fax. 700261 http://darmajaya.ac.id

FORMULIR

BIRO ADMINISTRASI AKADEMIK KEMAHASISWAAN (BAAK)

FORM KONSULTASI/BIMBINGAN SKRIPSI/TUGAS AKHIR *)

NAMA : Diki Datri Mulyoto
 NPM : 1911010054
 PEMBIMBING I : Suliyana S.tam., M.T.
 PEMBIMBING II :
 JUDUL LAPORAN : Face Recognition Presensi Mahasiswa IIB Darmajaya
Menggunakan Metode Eigenface
 TANGGAL SK :s.d..... (6+2 bulan)

No	HARI/TANGGAL	HASIL KONSULTASI	PARAF
1	25/22 /10	Penentuan metode penditran untuk judul skripsi	
2	31/22 /10	Bimbingan proposal bab 1-3 Menentukan variabel penditran	
3	1/22 /11	Penelitian terkait menunjang riset yang diteliti	
4	22/22 /11	Perbaikan bab 1-3 dan Acc sidang	
5	6/22 /12	konsultasi terkait sidang sempro materi PPT	
6	31/22 /01	Pengarahan aplikasi sebelum pengujian aplikasi	
7	8/23 /02	Perbaikan metode aplikasi agar berjalan lebih baik	
8	10/23 /02	Bimbingan bab 4-5	
9	14/23 /02	Acc sidang skripsi	
10			

*) Coret yang tidak perlu

Bandar Lampung, 25 Nember 2023
Ketua Jurusan

Dr. Chairani, S.tam., M.Ts

Algoritma Eigenface

Dalam menjalankan program aplikasi menggunakan metode eigenface pertama-tama kita perlu melakukan *training* beberapa data wajah menggunakan *code* pada `train.py`.

```

14 class Train:
15     def __init__(self, config: Config) -> None:
16
17         self.config = config
18
19         self.init_config()
20
21     def init_config(self):
22         self.mean_vec = self.config.mean_vec
23         self.eig_vec = self.config.eig_vec
24         self.weights = self.config.weights
25
26         self.list_images = self.config.list_images
27         self.face_cascade = self.config.face_cascade
28
29     def add_image(self, id, frame):
30         root = 'real-time/Faces/%s/' % (id)
31         name = "%s.jpg" % hash(time.time())
32         try:
33             os.makedirs(root) # Making folder for a new face
34             cv2.imwrite(root + name, frame)
35         except:
36             cv2.imwrite(root + name, frame)
37
38         getface.getface(id)
39
40         images, len_images, size, images_path = model.images("real-time")
41         mean_vec = model.mean_vec(images, size, len_images)
42         nor_images = model.normalise(images, mean_vec)
43
44         k = (len(nor_images) / 2) + 1
45         eig_vec, weights = model.pca(nor_images, k) # Using PCA to get Eigenfaces
46
47         np.save("real-time/vectors/mean_vec", mean_vec) # Storing mean_vec for real time
48         np.save("real-time/vectors/eig_vec", eig_vec) # Storing eig_vec for real time

```

train.py

Langkah berikutnya dalam penggunaan metode eigenface adalah pendeteksian wajah dimana aplikasi dapat melakukan *tracking* wajah dan kemudian mengubah wajah menjadi *black and white* serta melakukan *crooping* dan *saving* data dengan *code* `getface.py`.

```

def getface(myname): #Getting face from an image
    face_cascade = cv2.CascadeClassifier('ui/data/haarcascade_frontalface_alt2.xml')

    folder = 'real-time/Faces/%s/%s'%(myname)
    root = folder

    images_path = []
    pattern = "*.jpg"
    for path, subdirs, files in os.walk(root): #For all images in the 'myname' directory
        for name in files:
            if fnmatch(name, pattern):
                img_root = root + name
                img = cv2.imread(img_root)
                gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY) #Color to B&W
                faces = face_cascade.detectMultiScale(gray, 1.2, 5) #Detecting faces, scaleFactor = 1.3, neighbours = 5
                for (x,y,w,h) in faces:
                    face_img = img[y+5:y+h-5,x+5:x+w-5] #Cropping face image
                    face_img = cv2.resize(face_img, (32,32)) #Resizing to make uniform images
                    cv2.imwrite(img_root, face_img) #Saving the image back

    print("%s's Face Extracted\n" % (myname))

```

getface.py

Untuk melakukan *saving* data maka data akan diproses dan diberikan folder dari masing-masing wajah yang sudah di training seperti nama mahasiswa dan npm mahasiswa tersebut menggunakan code datawajah.py.

```

    uic.loadUi("ui/data_wajah.ui", self)

    self.disply_width = 640
    self.display_height = 480
    self.rgb_image = None
    self.id = None

    # allow only integers
    onlyInt = QtGui.QIntValidator()
    onlyInt.setRange(0, 100)
    self.kameraIndexEdit.setValidator(onlyInt)

    self.tutupButton.clicked.connect(self.close)
    self.mulaiButton.clicked.connect(self.mulaiButtonClick)
    self.simpanWajahButton.clicked.connect(self.simpanWajahButtonClick)

    self.config = Config()
    self.algo = Eigenfaces(config=self.config)
    self.train = Train(self.config)

    self.initVideoThread()

    def simpanWajahButtonClick(self):
        if self.id is None and self.rgb_image is None:
            QtWidgets.QMessageBox.critical(
                self, "Simpan image", "Gagal simpan, image tidak valid"
            )
            return

        self.train.add_image(self.id, self.rgb_image)

        QtWidgets.QMessageBox.information(
            self,
            "Simpan image",
            "Berhasil disimpan",

```

datawajah.py

Wajah mahasiswa yang telah di *training* dan disimpan sebelumnya mengalami perhitungan nilai *mean* dan nilai PCA untuk berikutnya dapat melakukan identifikasi wajah yang telah di *training* menggunakan code `model.py`.

```
def images(dataset):
    if dataset == "preset": # For accuracy on a dataset
        folder = "/preset/"

    elif dataset == "real-time": # For real-time
        folder = "/real-time/"

    in_path = os.path.realpath("train.py")
    root = os.path.dirname(in_path) + folder

    pattern = "*.jpg"

    images_path = []

    for path, subdirs, files in os.walk(root): # collecting all images for training
        for name in files:
            if fnmatch(name, pattern):
                images_path.append(os.path.join(path, name))

    images = []
    for img_path in images_path:
        img = cv2.imread(img_path, cv2.IMREAD_GRAYSCALE) # Grayscale input
        for_img_size = img

        img = img.flatten() # Flatten to M*N*1 vector
        images.append(img)

    size = for_img_size.size # size of image

    return images, len(images), size, images_path

def mean_vec(images, size, m): # Getting mean_vec of training set
    sum_input = [0 for i in range(size)]
    for img in images:
```

`model.py`

Setelah langkah-langkah diatas terpenuhi maka kita dapat melakukan identifikasi wajah yang telah melakukan *training* berupa data wajah, nama dan npm mahasiswa menggunakan `code eigenface.py`.

```

class Eigenfaces:
    def __init__(self, config: Config):
        self.config = config
        self.frame = None
        self.init_config()

    def faces(self): # method ini mengambil frame dari kamera, mengkonversikan frame menjadi grayscale,
                    # dan melakukan deteksi wajah.
        if self.frame is None:
            return None
        gray = cv2.cvtColor(self.frame, cv2.COLOR_BGR2GRAY) # Color ot B&W
        faces = self.face_cascade.detectMultiScale(gray, 1.2, 5) # Detecting Face
        return faces

    def detek(self): # method ini mengambil wajah yang terdeteksi oleh method faces dan melakukan pengenalan wajah
        faces = self.faces()
        if faces is None:
            return None

        ids = []
        for (x, y, w, h) in faces:
            face_img = self.frame[y_: y + h, x_: x + w]
            face_img = face_img[:, :, 0]
            face_img = cv2.resize(face_img, (32, 32))

            ind, dis = test.test_img(
                self.mean_vec, self.eig_vec, self.weights, face_img
            ) # Checking the image class
            if dis != -1:
                list_images = list(
                    pd.read_csv("real-time/vectors/image_path.csv")["0"]
                ) # For Indexing

                directory = pathlib.PurePath(os.path.dirname(list_images[ind]))
                text_name = directory.name # For Name of the face class

```

eigenfaces.py



Plagiarism Checker X Originality Report

Similarity Found: 10%

Date: Friday, February 17, 2023

Statistics: 468 words Plagiarized / 4893 Total words

Remarks: Low Plagiarism Detected - Your Document needs Optional Improvement.

BAB I PENDAHULUAN Latar belakang Perkembangan teknologi saat ini sudah menjangkau banyak bidang. Dengan kemajuan ini maka banyak terjadi perubahan, dimana saat ini kita menghadapi revolusi penerapan digitalisasi teknologi. Dengan adanya Ponsel Pintar (smartphone) besar kemungkinan untuk menerapkan salah satu teknologi dengan memanfaatkan perkembangan teknologi digital. Ada berbagai macam sistem pencatatan kehadiran mahasiswa sebagai bukti bahwa mahasiswa sudah berada dalam kelas sesuai mata kuliah.

Pada Jurusan Teknik Informatika IIB Darmajaya, sistem pencatatan kehadiran mahasiswa dilakukan secara manual baik melalui tanda tangan atau panggilan lisan oleh dosen mata kuliah. Hal tersebut membuat banyak celah bagi mahasiswa untuk melakukan tindak kecurangan. Mahasiswa dapat melakukan pemalsuan tanda tangan dimana mahasiswa tidak masuk mata kuliah tersebut tetapi absennya dapat diisi oleh teman. Oleh karena itu, dibutuhkan sistem yang dapat meminimalisir kecurangan dalam pencatatan kehadiran mahasiswa di dalam kelas dengan melakukan verifikasi.

Verifikasi adalah proses yang diperlukan di mana keakuratan data atau informasi diperlukan, misalnya saat mengkonfirmasi kebenaran suatu identifikasi. Perusahaan, birokrasi, dan lainnya telah menggunakan verifikasi ini secara ekstensif seiring dengan kemajuan teknologi dan informasi. Menggunakan detektor sidik jari (fingerprint) adalah metode umum untuk memverifikasi identitas. Kurangnya fleksibilitas dalam verifikasi ini adalah suatu kelemahan dari aplikasi ini. Teknologi yang digunakan tidak dapat diakses dimanapun atau diakses setiap saat, yang membatasi fleksibilitas.

Oleh karena itu, pada tugas akhir ini akan digunakan sistem absensi berbasis website. Sistem absensi saat ini diproyeksikan akan lebih fleksibel setelah penerapan kedalam