1. **Kompetensi**

Mahasiswa mampu membuat, memindai, dan mengestrak informasi dari Aruco Marker/AR Tag dengan library OpenCV.

1. **Sub Kompetensi**

* Dapat melakukan pemrograman dengan Jupyter Notebook.
* Dapat melakukan pemrograman membuat Aruco Marker.
* Dapat melakukan ekstrasi informasi pada Aruco Marker.

1. **Dasar** **Teori**

**Aruco Marker**

Chart

Description automatically generated with medium confidence

MarkerArUco adalah marker kotak sintetik yang disusun oleh garis tepi hitam lebar dan matriks biner bagian dalam yang menentukan pengidentifikasi (id). Perbatasan hitam memfasilitasi deteksi cepat dalam gambar dan kodifikasi biner memungkinkan identifikasi dan penerapan teknik deteksi kesalahan dan koreksi. Ukuran marker menentukan ukuran matriks internal. Misalnya ukuran marker 4x4 terdiri dari 16 bit. Manfaat utama mereka adalah deteksi mereka kuat, cepat dan sederhana. Modul ArUco mencakup deteksi jenis penanda ini dan alat untuk menggunakannya untuk pose estimation dan kalibrasi kamera. Untuk fungsinya dapat digunakan untuk pengenalan objek dalam misi pendaratan dan pembacaan jarak dengan menggunakan library ArUco Marker yang terdapat pada OpenCV, sehingga didapatkan sumbu x, y dan z dari Markers.

1. **Alat dan Instrumen**

* Komputer (PC/Laptop)
* Browser (Chrome/Edge/Firefox)
* OpenCV Contrib
* ArUco Marker
* Jupyter Notebook

1. **Langkah Kerja**
2. **Mengimport library yang dibutuhkan**
3. Jalankan Anaconda Navigator **as Administrator**

Graphical user interface, application

Description automatically generated

1. Untuk memulai pemrograman Python dengan Jupyter Notebook, silahkan Launch Jupyter Notebook

Graphical user interface, application

Description automatically generated

1. Setelah di **Launch** maka akan menuju browser default yang di gunakan pada masing-masing device.

A screenshot of a computer

Description automatically generated

1. Untuk memulai menjalankan Script Python, silahkan klik tombol **New** dan arahkan ke Python 3 seperti gambar berikut.

Graphical user interface, application

Description automatically generated

1. Lalu jendela baru akan terbuka untuk membuka text editor kernel Jupyter Notebook

Graphical user interface, application, Word

Description automatically generated

1. Kemudian import library berikut:

Graphical user interface, text, application

Description automatically generated

import cv2 as cv

import cv2.aruco as aruco

import numpy as np

import os

1. Klik **Run** pada Jupyter Notebook.

Graphical user interface, application

Description automatically generated

1. **Mengenerate Aruco Marker**
2. Ketikan kode berikut untuk mengenerate/membuat qrcode dari sebuah link situs.

Graphical user interface, text, application

Description automatically generated

# Load the predefined dictionary

dictionary = cv.aruco.Dictionary\_get(cv.aruco.DICT\_6X6\_250)

# Generate the marker

markerImage = np.zeros((200, 200), dtype=np.uint8)

markerImage = cv.aruco.drawMarker(dictionary, 33, 200, markerImage, 1);

cv.imwrite("marker33.png", markerImage);

1. Lalu klik **Run**
2. Silahkan cek pada directory jupyter notebook tersebut maka akan tergenerate Aruco Marker yang dibuat, seperti berikut

A picture containing graphical user interface

Description automatically generated

1. **Membuat program untuk scan Aruco Marker**
2. Setelah dibuatnya/generate Aruco Tag/Aruco Marker maka selanjutnya akan dibuat program untuk melakukan scan Aruco Tag/Aruco Marker.
3. Ketik code berikut ini untuk membuat **function generating Marker**

Graphical user interface, text, application

Description automatically generated

def findArucoMarkers(img, markerSize = 4, totalMarkers=250, draw=True):

gray = cv.cvtColor(img, cv2.COLOR\_BGR2GRAY)

key = getattr(aruco, f'DICT\_{markerSize}X{markerSize}\_{totalMarkers}')

arucoDict = aruco.Dictionary\_get(key)

arucoParam = aruco.DetectorParameters\_create()

bboxs, ids, rejected = aruco.detectMarkers(gray, arucoDict, parameters = arucoParam)

print(ids)

if draw:

aruco.drawDetectedMarkers(img, bboxs)

cap = cv.VideoCapture(0)

while True:

success, img = cap.read()

findArucoMarkers(img)

cv2.imshow('img',img)

k = cv2.waitKey(30) & 0xff

if k == 27:

break

cap.release()

cv2.destroyAllWindows()

**Note: Apabila memiliki webcam eksternal silahkan ubah cv2.VideoCapture(0) menjadi cv2.VideoCapture(1)**

1. Kemudian klik **Run**
2. Sebelum di **Run** pastikan webcam laptop/pc mu tersedia dan bisa digunakan.
3. Berikut akan muncul pop up ketika program dijalankan lalu arahkan untuk menscan ArTag/Aruco Marker

A screenshot of a computer

Description automatically generated with medium confidence

1. Terlihat pada output Jupyter Notebook dan tampilan scanner terlihat ektraksi informasi yang terdapat dalam Aruco Marker yaitu id .
2. Untuk keluar dari program silahkan klik pop up yang muncul lalu klik di keyboard ‘esc’.
3. Apabila terjadi not responding silahkan restart kernel pada Jupyter Notebook seperti berikut.

Graphical user interface, application

Description automatically generated

1. **Tugas**
   1. Buatlah program *generate Aruco Marker* dengan bedakan *input* datanya!
   2. Buatlah program *do-something* setelah melakukan *scan Aruco Marker*!
   3. Buatlah program untuk *scan Aruco Marker* dengan berbeda ukuran *pixel*, lalu analisalah!