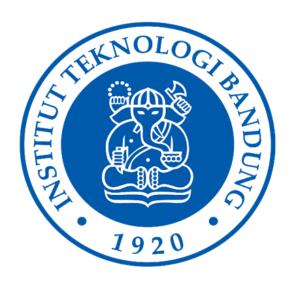
## Milestone B IF4074 Pembelajaran Mesin Lanjut

**Convolutional Neural Network** 



#### Disusun Oleh:

Farras Mohammad Hibban Faddila

13518017

# TEKNIK INFORMATIKA SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA INSTITUT TEKNOLOGI BANDUNG SEMESTER 1 2023/2024

### 1. Penjelasan Kode Program

Terdapat beberapa kelas yang strukturnya mencoba mengimitasi kelas-kelas pada library Tensorflow, yang kemudian diimplementasi sendiri, seperti:

- Dari milestone pertama, terdapat beberapa fungsi tambahan yang dibuat:
- 1. Penyimpanan model dan load model dari file eksternal

```
def save model(self):
     params = []
     for layer in self.layers:
       params.append(layer.to object())
    with open("model.json", "w") as f:
       data = json.dumps(params)
       f.write(data)
 def load model(self, filepath):
     layers from file = []
    with open(filepath, "r") as f:
       data = json.load(f)
     for layer in data:
       print(layer["type"])
       layer obj = None
       if layer["type"] == "conv2d":
          layer obj = Convolutional(
             input shape=layer["input shape"],
             padding=layer["padding"],
            filter count=layer["filter count"],
            kernel shape=layer["kernel shape"],
            stride=laver["stride"].
       elif layer["type"] == "dtctr":
          layer obj = Detector(activation=layer["activation"])
       elif layer["type"] == "max pool":
          layer obj = Pooling(size=layer["size"], stride=layer["stride"], mode="max")
       elif layer["type"] == "avg pool":
          layer obj = Pooling(size=layer["size"], stride=layer["stride"], mode="avg")
       elif layer["type"] == "flatten":
          laver obj = Flatten()
       elif layer["type"] == "dense":
          layer obj = Dense(size=layer["size"], input size=layer["input size"],
activation=layer["activation"])
       layers from file.append(layer obj)
```

```
self.layers = layers from file
```

#### 2. Backpropagation

```
def backward(self, din, learning rate):
   (num channels, orig dim) = self.last input.shape
   dout = np.zeros(self.last input.shape)
  for c in range(num_channels):
     tmp y = out y = 0
     while tmp_y + self.size <= orig_dim:
        tmp x = out x = 0
        while tmp x + self.size \le orig dim:
          patch = self.last input[
             c, tmp_y : tmp_y + self.size, tmp_x : tmp_x + self.size
          (x, y) = np.unravel_index(np.nanargmax(patch), patch.shape)
          dout[c, tmp y + x, tmp x + y] += din[c, out y, out x]
          tmp x += self.stride
          out x += 1
        tmp_y += self.stride
        out y += 1
   return dout
```

# 2. Pembagian Kerja

Nama	Pekerjaan
Farras Mohammad Hibban Faddila	Seluruhnya