



# Outline

- **Dataset**
- **Handcraft-Based Approach**
- **Learning-Based Approach**

• • • • • • •

# Dataset



• • • • •

# Dataset

เรามีวิธีการประมวลผลภาพเพื่อให้คุณลักษณะมีความชัดเจนมากยิ่งขึ้นโดยใช้วิธีการ เช่น

- Canny
- Erosion
- Dilation
- Opening
- Closing

Dataset ที่ใช้ทดสอบทั้งหมดประมาณ 3 หมื่น โดยมีขนาดภาพที่แตกต่างกันไป มี Object หลักคือใบหน้าคน

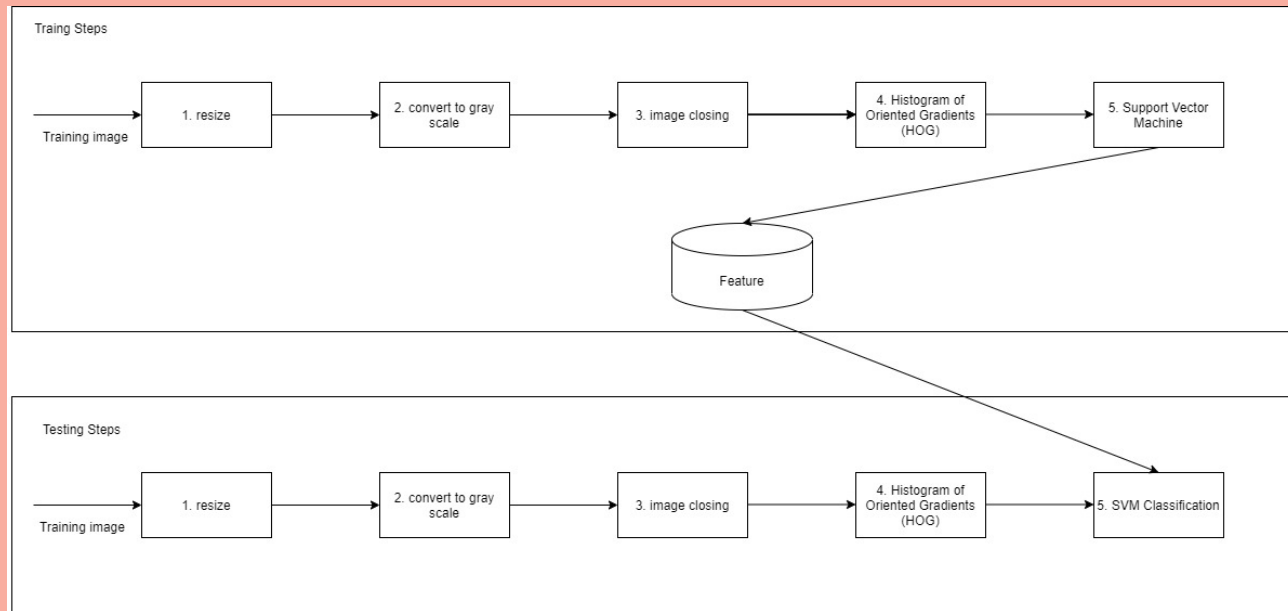


# Handcraft-Based Approach



• • • • •

# Diagram



# การประเมินคุณลักษณะ

วิธีการประเมินประสิทธิภาพที่เลือกใช้ คือ ซัพพอร์ตเวกเตอร์แมชชีน (SVM) โดยช่วงของการ train ให้ทำการคำนวณผลและฝึกฝนไปก่อน เมื่อนำข้อมูลมาทำการ test ข้อมูลของตัวที่ test จะทำการเรียนรู้จากตัวที่เคยผ่านการฝึกฝนมาก่อนหน้านี้



# Test โดยทำ Image Processing



Gray Scale

• • • • • • •



## หาเส้นขอบโดยใช้ Canny



Gray scale



Result

```
human_face
```

```
Success Rate: 93.22
```

```
Failure Rate: 6.78
```

# Result



# Morphological



• • • • •

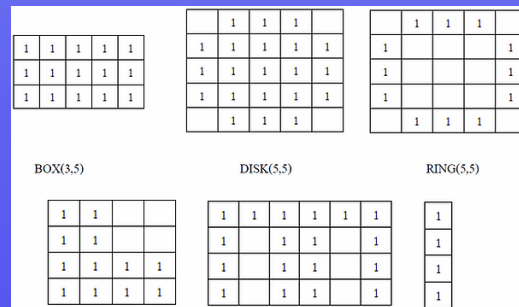
# Morphological

**Morphological Operation** คือ เซตของการประมวลผลทางภาพในเรื่องรูปร่าง โดยการประยุกต์ Structure Element มา Operate กับ Input image และได้ผลลัพธ์หรือ Output image ที่มีขนาดเท่ากับ Input image

**Structure Element** คือ เมตริกที่ถูกนิยามให้เป็นรูปร่างและขนาดที่เป็น Neighborhood สำหรับทำ Morphological Operation โดยในเมตริกจะประกอบด้วยค่า Binary 2 ค่า คือ 0 และ 1 ซึ่งสามารถมีรูปร่างตามที่เรากำหนด โดยที่ 1 จะกำหนดเป็น Neighborhood ดังรูปด้านล่าง

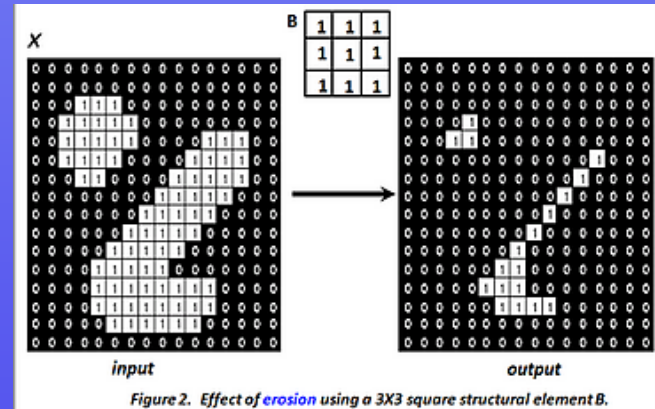
## Basic Operation

- Erosion
- Dilation
- Opening
- Closing



# Erosion

เป็นการย่อภาพในลักษณะของการลบข้อมูลภาพบริเวณขอบ โดยทั้งหมดที่อยู่ใกล้เคียงนั้นจะโดนลบออกไปตามขนาดของ kernel ดังนั้นความหนาหรือขนาดของวัตถุข้างหน้าจะลดลง และสิ่งนั้นจะเป็นประโยชน์ด้านการลด noises ออก



## ทำ Erosion ก่อนแล้วใช้ Canny



**Gray scale**



**Erosion**



**Result**

```
human_face
```

```
Success Rate: 93.22
```

```
Failure Rate: 6.78
```

# Dilation

เป็นการขยายภาพในลักษณะของการเพิ่มข้อมูลภาพตามลำดับตลอดทั้งภาพ โดยจะเป็นเพิ่มส่วนสีขาวหรือขนาดของวัตถุเบื้องต้น ปกติในกรณีการลด Noise นั้นจะใช้ Erosion ขยายและลดภาพสีขาวลง และจะทำให้วัตถุหดตัวลง โดยวิธีนี้เหมาะสำหรับการเพิ่มจุดที่เสียหายไป

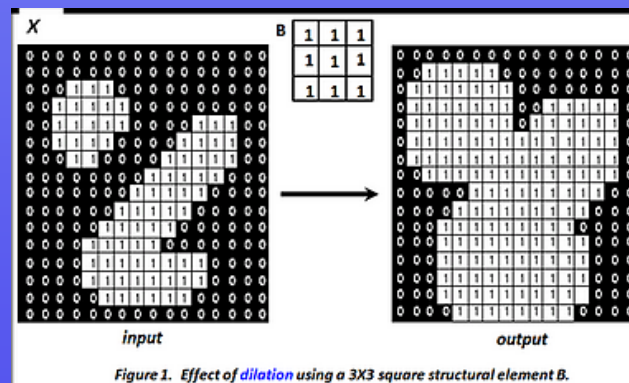


Figure 1. Effect of dilation using a 3X3 square structural element B.

## ทำ Dilation ก่อนแล้วใช้ Canny



**Gray scale**



**Dilation**



**Result**

```
human_face
```

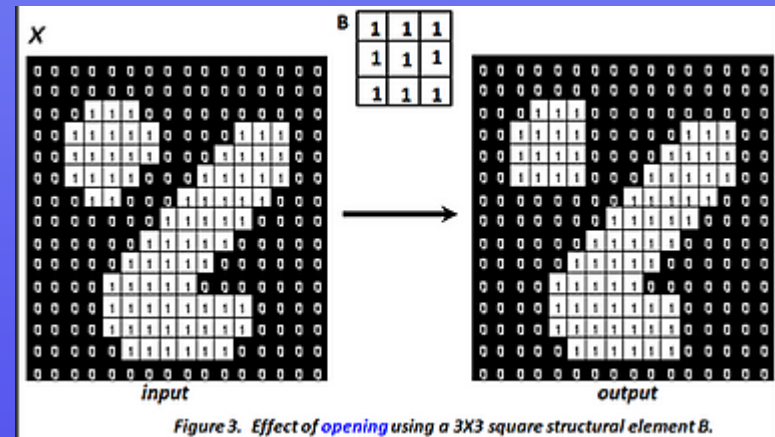
```
Success Rate: 94.54
```

```
Failure Rate: 5.46
```



# Opening

นิยามของ opening คือเอา image มา erode แล้วค่อย dilate ใช้ในการลบ noise เพราะว่า noise หายไปตอน erode แต่ขนาดของวัตถุเล็กลงก็เอาคืนด้วยการ dilate ใช้ในการลบขอบที่ยื่นๆ ของวัตถุด้วย



## ทำ Opening ก่อนแล้วใช้ Canny



**Gray scale**



**Opening**



**Result**

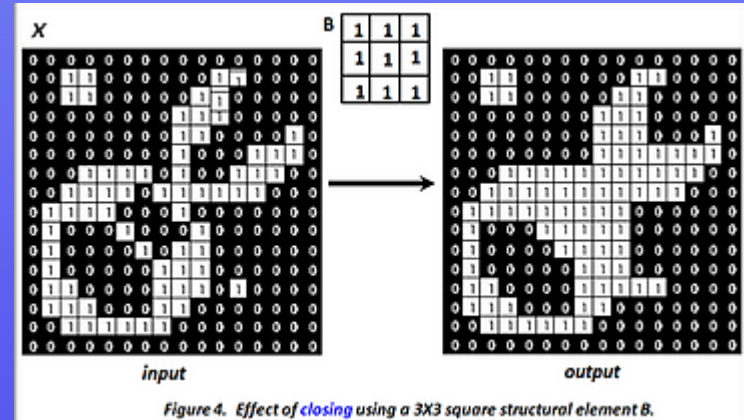
human\_face

Success Rate: 92.42

Failure Rate: 7.58

# Closing

ตรงข้ามกับ opening คือการนำ image มา dilate แล้ว erode ใช้ในการลบ small holes สามารถใช้ในการ เชื่อมวัตถุที่แยกจากกัน



## ทำ Closing ก่อนแล้วใช้ Canny



**Gray scale**



**Closing**



**Result**

```
human_face  
Success Rate: 95.51  
Failure Rate: 4.49
```

# Learning-Based Approach



• • • • •

# GoogLeNet

ใน Project นี้เราเลือกใช้ structure คือ GoogLeNet

โดย GoogLeNet เป็นโครงข่ายประสาทเทียมแบบ deep convolutional 22 ชั้นซึ่งเป็นรูปแบบหนึ่งของ Inception Network

GoogLeNet ประกอบด้วย 22 เลเยอร์ (27 เลเยอร์รวมเลเยอร์รวมกัน) และส่วนหนึ่งของเลเยอร์เหล่านี้เป็นโมดูลเริ่มต้นทั้งหมด 9 โมดูล GoogLeNet เป็นสถาปัตยกรรมหลักภายในไลบรารี ML ทั่วไป

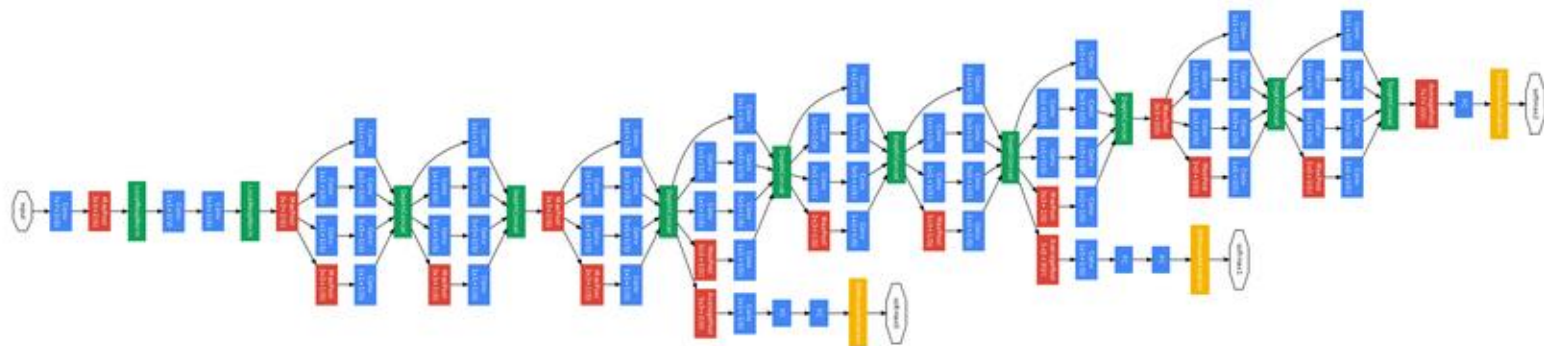
• • • • • • •



# Structure

• • • • • • •

# Structure



• • • • • • •



# Data augmentation

```
PRETRAINED_SIZE = 224 # define pretrained size
PRETRAINED_MEANS = [0.485, 0.456, 0.406] # define pretrained means
PRETRAINED_STDS = [0.229, 0.224, 0.225] # define pretrained stds
TRAIN_TRANSFORMS = transforms.Compose([ # define transforms for train datasets
    transforms.Resize(PRETRAINED_SIZE),
    transforms.RandomRotation(5),
    transforms.RandomHorizontalFlip(0.5),
    transforms.RandomCrop(PRETRAINED_SIZE, padding=10),
    transforms.ToTensor(),
    transforms.Normalize(mean=PRETRAINED_MEANS, std=PRETRAINED_STDS)])
TEST_TRANSFORMS = transforms.Compose([ # define transforms for test datasets
    transforms.Resize(PRETRAINED_SIZE),
    transforms.CenterCrop(PRETRAINED_SIZE),
    transforms.ToTensor(),
    transforms.Normalize(mean=PRETRAINED_MEANS, std=PRETRAINED_STDS)])
BATCH_SIZE = 40 # define batch size
```



# Result



```
Accuracy for class buildings is: 92.4 %  
Accuracy for class forest is: 98.5 %  
Accuracy for class glacier is: 87.0 %  
Accuracy for class human_face is: 100.0 %  
Accuracy for class mountain is: 83.0 %  
Accuracy for class sea is: 93.9 %  
Accuracy for class street is: 91.2 %
```

# สมาชิก



นายชวิน โล่ห้รัตนเสนห์  
62070045



นายณัฐวัฒน์ สามสี  
62070067



นายเดชพนต์ นุ่นเสน  
62070070



นายธีรภัทร์ บุญช่วยแล้ว  
62070096

