Fast Facial emotion recognition Using CNNs and Gabor Filters

RDBMS IA1

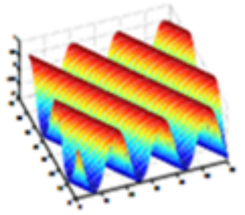
1911001-Aakash Saroop

Introduction

- *A deep learning based framework for human emotion detection is presented .*
- *Psychological theory states that human emotions can be classified into six different forms: surprise, fear, hatred, anger, happiness and sadness.*
- *The dataset JAFFE database is used.*

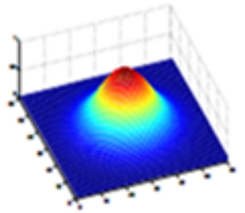
Prerequisites - Gabor Filter

- *Gabor filters are generally used in edge detection and feature extraction.*
- *A Gabor filter can be viewed as a sinusoidal signal of particular frequency and orientation, modulated by a Gaussian wave.*



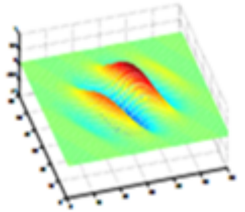
(a)

A Sinusoid oriented 30^0 with X-axis



(b)

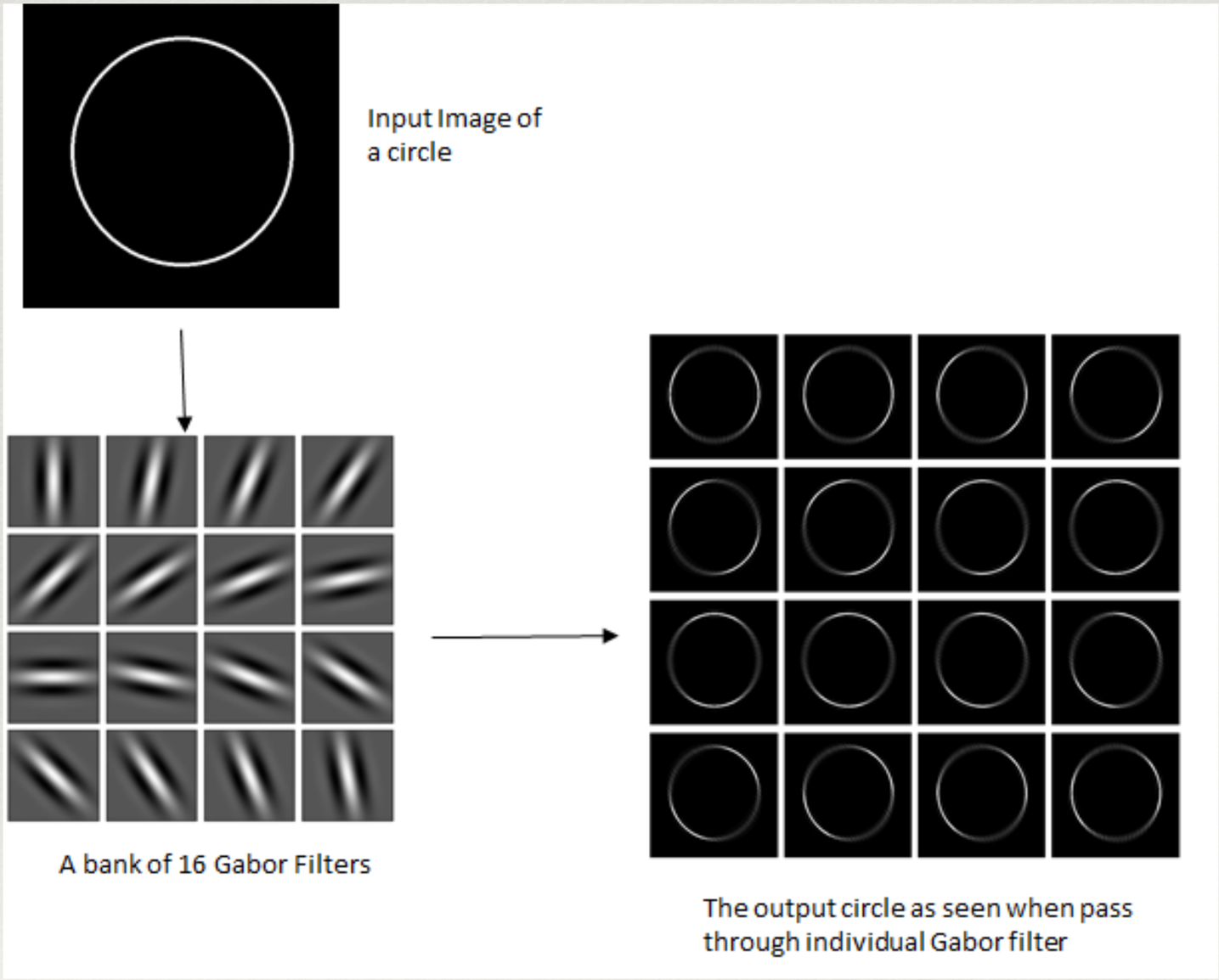
A 2-D Gaussian



(c)

The corresponding 2-D Gabor filter

- *It allows certain frequencies to pass and rejects others.*
- *When a Gabor filter is applied to an image, it gives the highest response at edges and at points where texture changes*


Input Image of
a circle
A bank of 16 Gabor Filters
The output circle as seen when pass
through individual Gabor filter

Complex

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \exp\left(i\left(2\pi\frac{x'}{\lambda} + \psi\right)\right)$$

Real

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi\frac{x'}{\lambda} + \psi\right)$$

Imaginary

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \sin\left(2\pi\frac{x'}{\lambda} + \psi\right)$$

where

$$x' = x\cos\theta + y\sin\theta$$

and

$$y' = -x\sin\theta + y\cos\theta$$

λ — Wavelength of the sinusoidal component.

θ — The orientation of the normal to the parallel stripes of Gabor function.

Ψ — The phase offset of the sinusoidal function.

σ — The sigma/standard deviation of the Gaussian envelope

γ — The spatial aspect ratio and specifies the ellipticity of the support of Gabor function.

First Gabor Filter

$(x, y) = (18,18),\ \sigma = 1.5,\ \theta = \pi/4,\ \lambda = 5,\ \gamma = 1.5$
$\psi = 0$

Second Gabor Filter

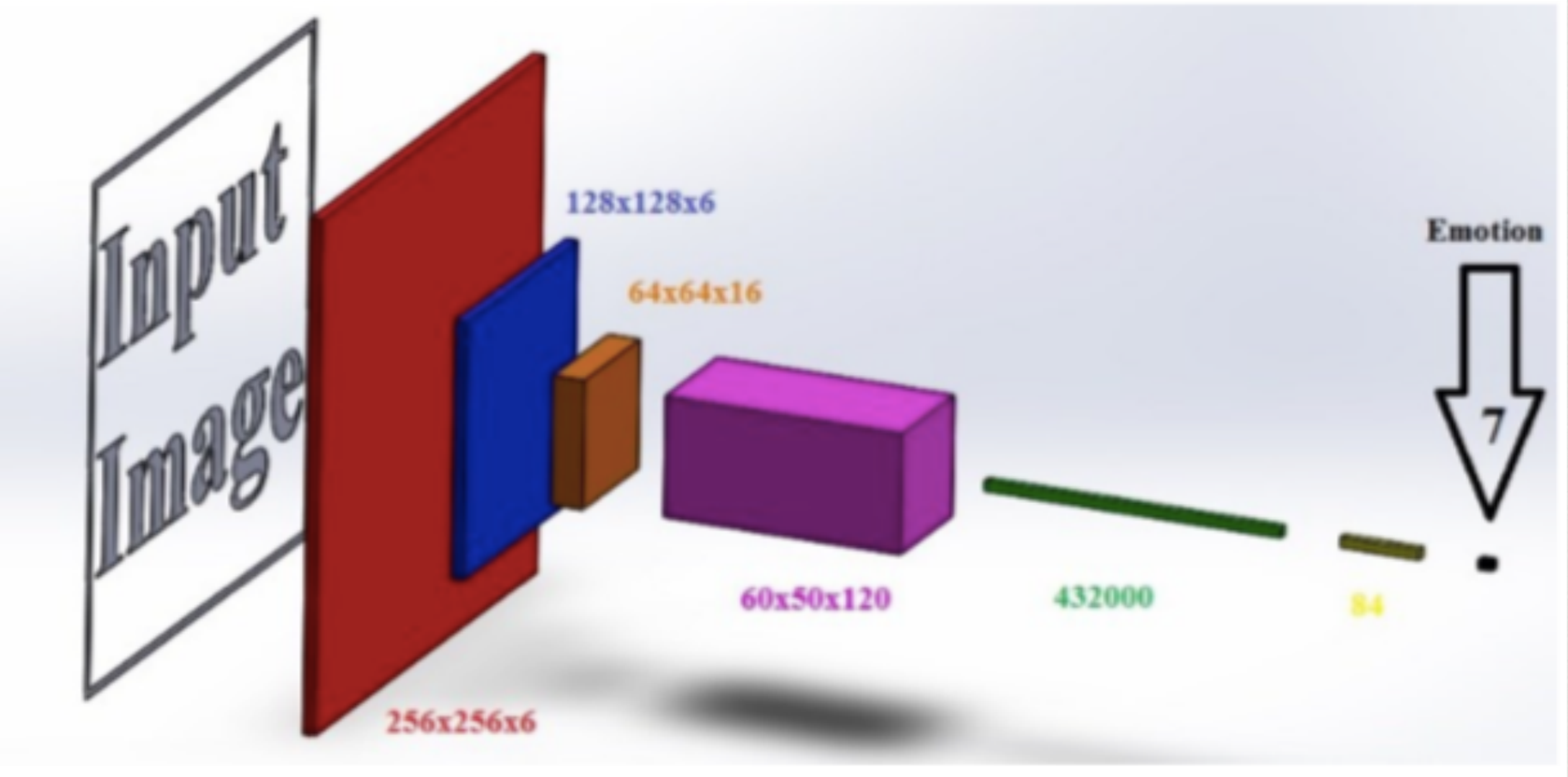
$(x, y) = (18,18),\ \sigma = 1.5,\ \theta = 3\pi/4,\ \lambda = 5,\ \gamma = 1.5$
$\psi = 0$

CNN model used

Table 1- Proposed CNN architecture method details

Layer type	Details	Output Shape
Conv	Conv (6x6)	256, 256, 6
Activation	Relu	256, 256, 6
MaxPooling	Pool size (2,2)	128, 128, 6
Conv	Conv (16x16)	128, 128, 16
Activation	Relu	128, 128, 16
MaxPooling	Pool size (2,2)	64, 64, 16
Conv	Conv (120x120)	60, 60, 120
Activation	Relu	60, 60, 120
Dropout	----------	60, 60, 120
Flatten	Flatten to a vector	432000
Dense	Input →84	84
Activation	Relu	84
Dropout	----------	84
Dense	Input → Classese Num =7	7
activation	softmax	7

CNN model used



Perfomance comparison

Epoch	CNN Methode Accuracy	2Gabor + CNN Methode
1	0.1492	0.1713
10	0.4696	0.5856
15	0.5996	0.7624
20	0.6519	0.8453
25	0.7790	0.8453
30	0.8232	0.9116

Conclusion

- *After applying the Gabor Filter, the system learning becomes faster and the accuracy has improved.*
- *This is because Gabor Filters extract image sub feature and gives the neural network.*

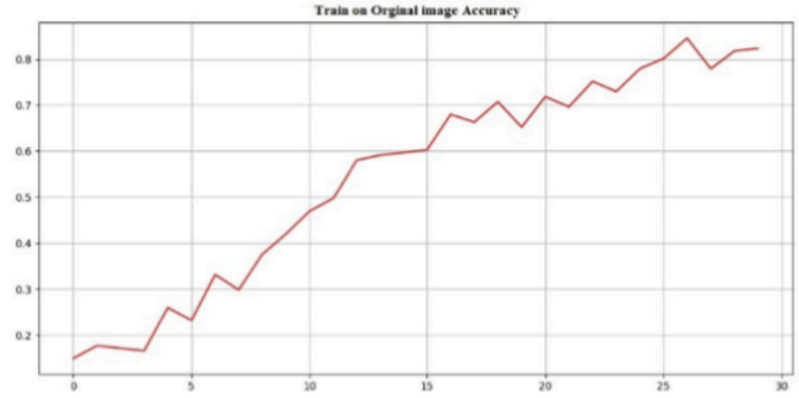


Figure 8 - Accuracy of simple CNN method

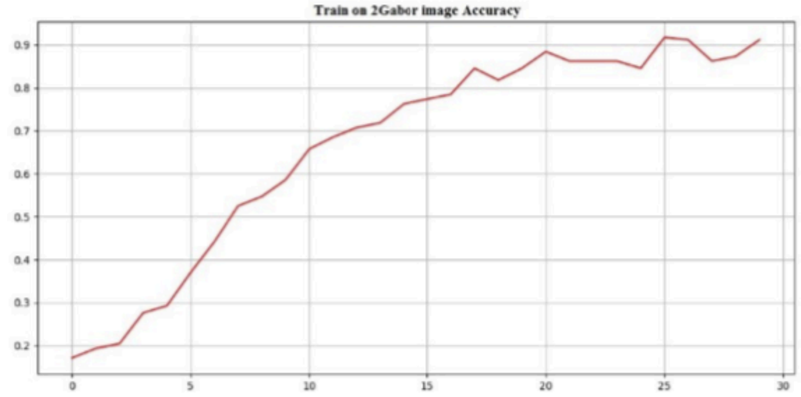


Figure 9 – Accuracy of 2Gabor filter+CNN method

Strategy for implementation

- *The dataset used in the research paper is extremely small, having only 213 images. While implementing I will use a larger dataset FER having ~35k images.*
- *Trying CNN models different to the one proposed in the research paper.*
- *Splitting the data into training and cross validation sets*
- *Using callbacks such as EarlyStopping and ReduceOnPlateau.*

References

- ***Research paper***: *https://www.researchgate.net/profile/Babak_Majidi/publication/344190368_Fast_Facial_emotion_recognition_Using_Convolutional_Neural_Networks_and_Gabor_Filters/links/5f9a4a7992851c14bcf08802/Fast-Facial-emotion-recognition-Using-Convolutional-Neural-Networks-and-Gabor-Filters.pdf*
- ***Gabor Filters***: *https://medium.com/@anuj_shah/through-the-eyes-of-gabor-filter-17d1fdb3ac97*
- ***CNN***: *https://en.wikipedia.org/wiki/Convolutional_neural_network*
- ***JAFFE dataset:*** *https://zenodo.org/record/3451524#.YFIjiy2B2fU*
- ***FER dataset:*** *https://www.kaggle.com/deadskull7/fer2013*


Thank You