

# Препознавање на емоции во слика/видео



Содержина

Вовед

Имплементација

Користени алгоритми

Резултати

# ЕМОЦИИ

- Емоциите се психички процеси што го изразуваат човековиот однос кон предметите, настаните и другите луѓе. Постојат 7 типови на основни емоции: гнев, среќа, одвратност, страв, неутралност, тага и изненадување.



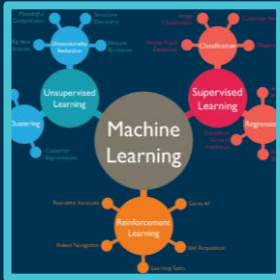


# Имплементација

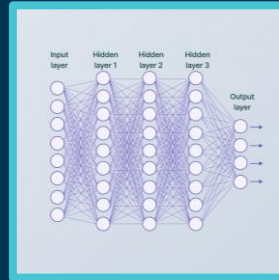
# Постапна имплементација



Анализа и  
средување на  
податочното  
множество



Одбирање на  
точниот модел



Тренирање на  
моделот



Препознавање на  
лице



Предвидување  
емоции

# Користени библиотеки

За имплементацијата на овој проект потребни беа Numpy, OpenCV, Tensorflow и Matplotlib библиотеците.



# Анализа и средување на податочното множество

- За да ни биде успешно тренирањето на моделот потребно е податоците на кои тој модел ќе го тренираме да бидат доволно добри и нормализирани

```
for features,label in  
training_data:
```

```
    X.append(features)
```

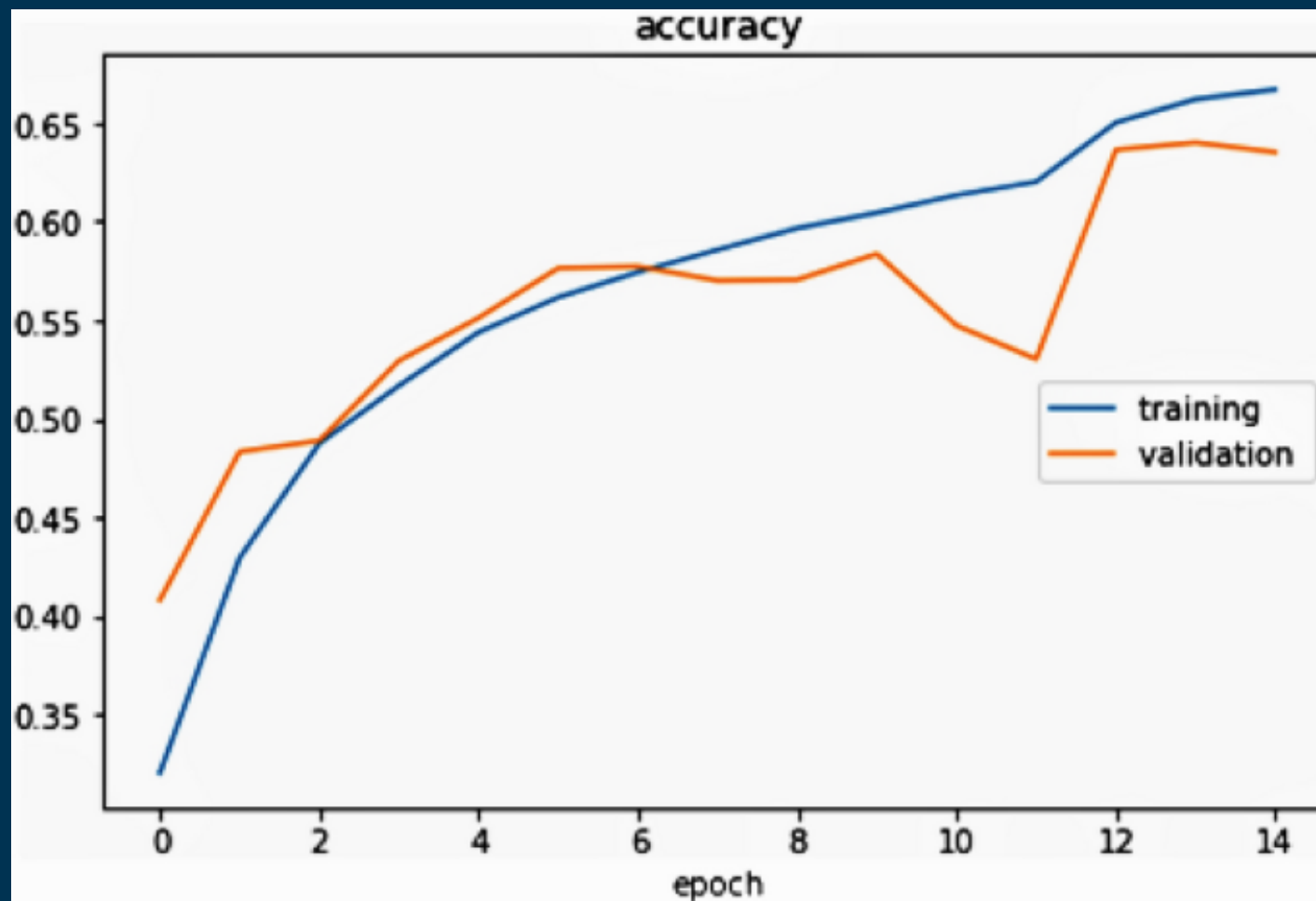
```
    Y.append(label)
```

```
X=np.array(X).reshape(-  
1,img_size,img_size,3)
```

```
X=X/255.0
```

# Одбирање на точниот модел

Поради големиот успех на FER 2013, за овој проект одбран е CNN(Convolutional Neural Networks) моделот, поточно MobileNetV2 моделот од Tensorflow библиотеката, се со цел да се добие највисока можна прецизност





# Тренирање на моделот

На моделот се поставуваат потребни слоеви, и потоа се тренира во пет епохи.

```
final_output=layers.Dense(128)(base_output)
```

```
final_output=layers.Activation('relu')(final_output)
```

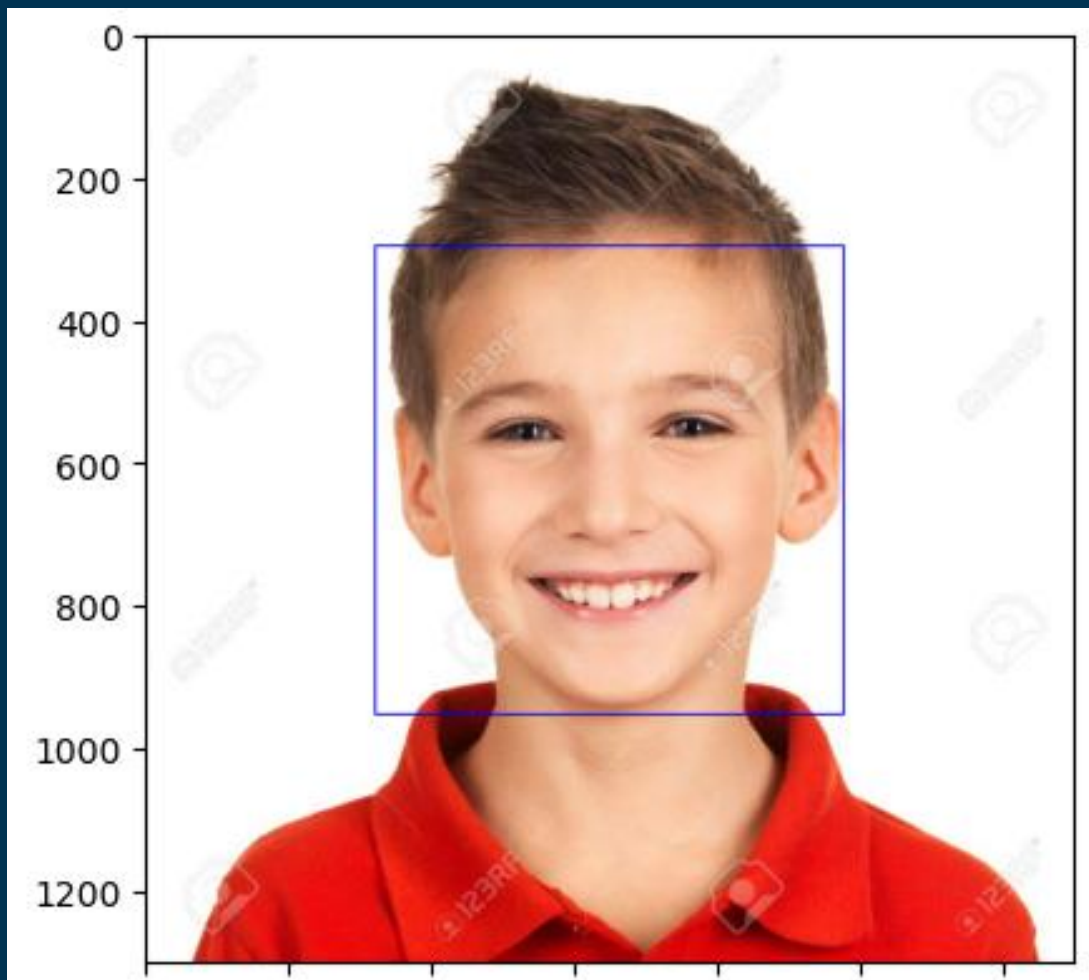
```
final_output=layers.Dense(64)(final_output)
```

```
final_output=layers.Activation('relu')(final_output)
```

```
final_output=layers.Dense(7,activation='softmax')(final_output)
```

```
new_model.fit(X,Y,epochs=5)
```

# Препознавање на лице




- Користејќи го HaarCascade методот принуден од openCV библиотеката успеваме да препознаеме лице релативно брзо

```
face_cascade=cv2.CascadeClassifier(cv2.data.haarcascades+'haarcascade_frontalface_default.xml')
faces=face_cascade.detectMultiScale(gray,1.1,4)
for x,y,w,h in faces:
    roi_gray=gray[y:y+h,x:x+w]
    roi_color=frame[y:y+h,x:x+w]
    cv2.rectangle(frame,(x,y),(x+w,y+h),(255,0,0),2)
    faces=face_cascade.detectMultiScale(roi_gray)
    if len(faces)==0:
        print("Face not detected")
    else:
        for(ex,ey,ew,eh) in faces:
            face_roi=roi_color[ey:ey+eh,ex:ex+ew]
            final_image=cv2.resize(face_roi,(224,224))
            final_image=np.expand_dims(final_image,axis=0)
```

# Препознавање на емоции

Користејќи го тренираниот модел и numpy библиотеката успешно предвидуваме емоции на слика/видео





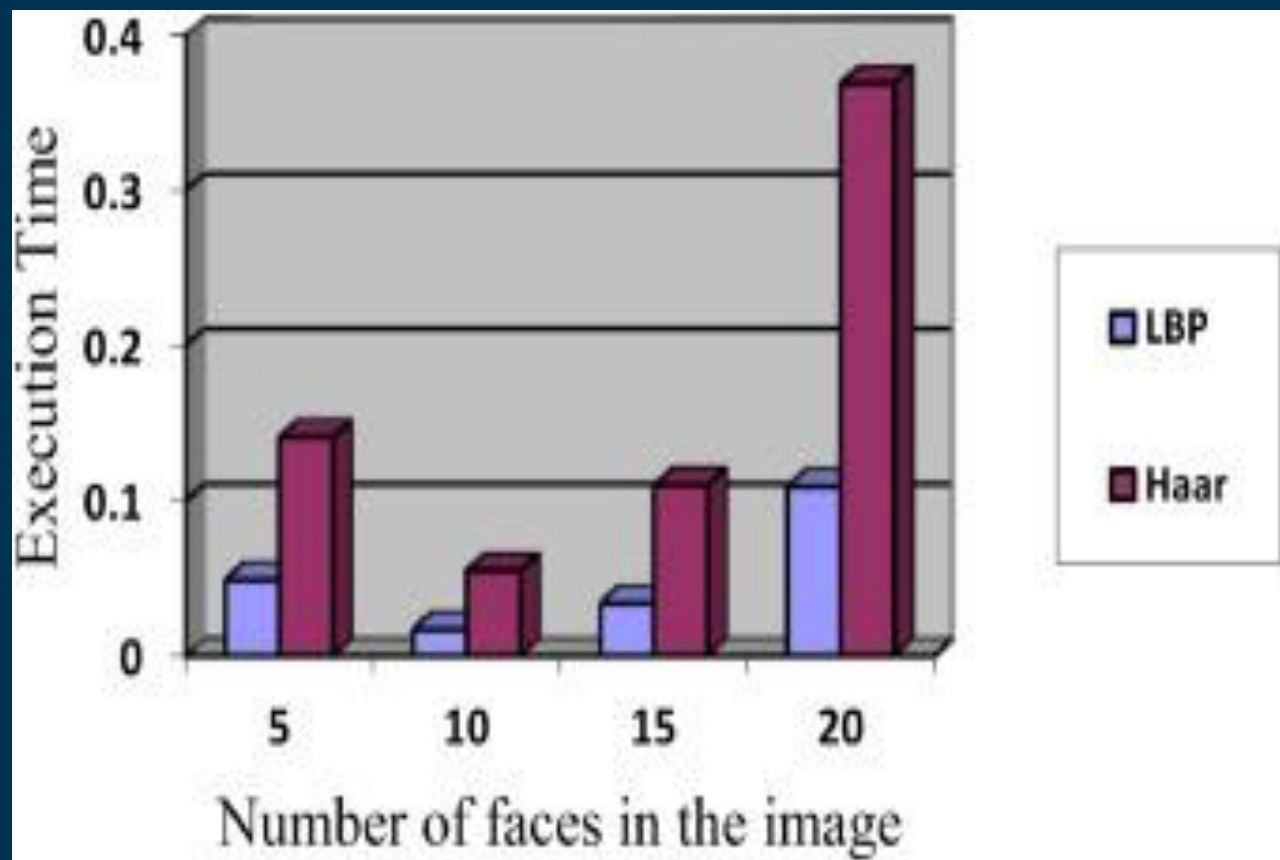
# Користени алгоритми

Haar  
cascade  
classifier



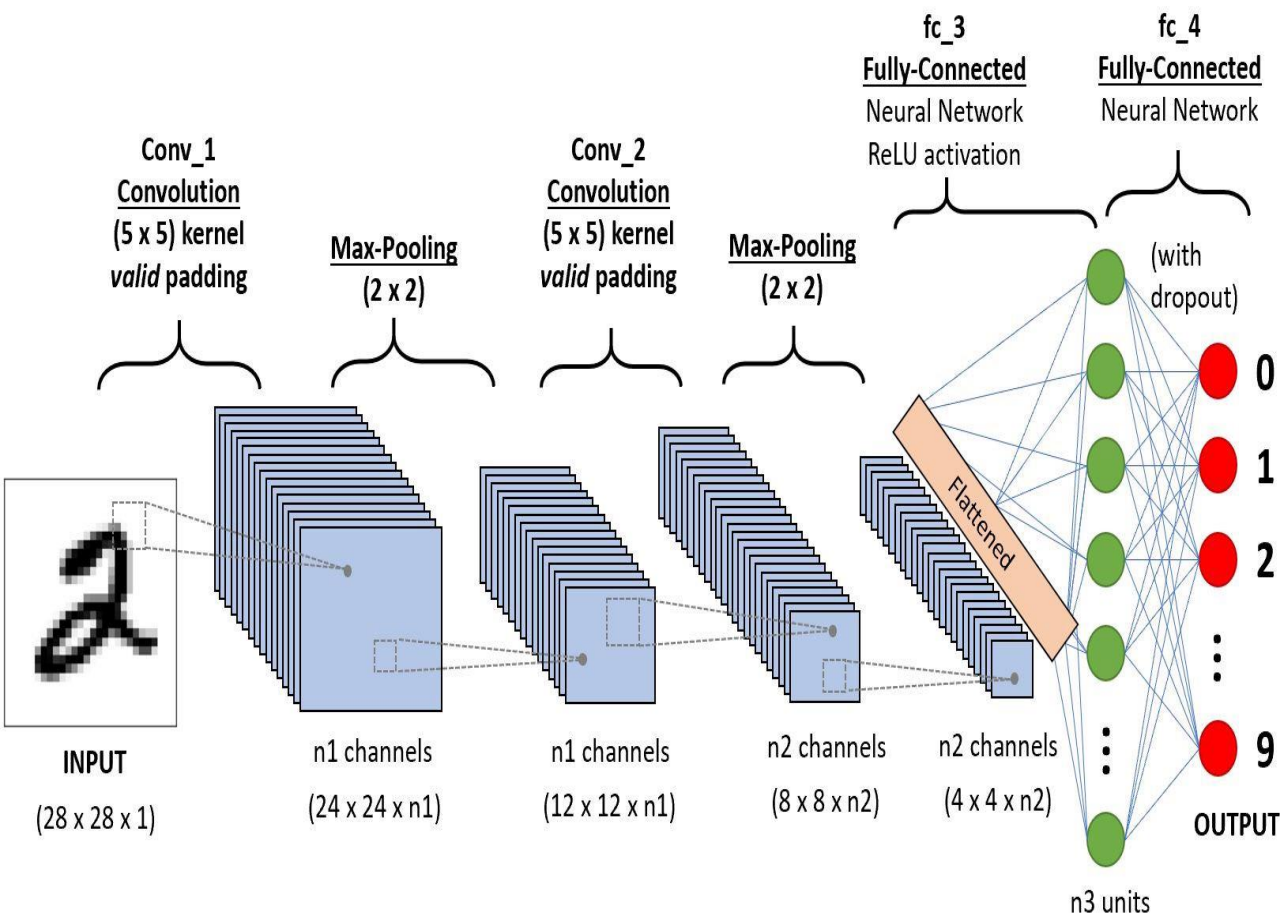
Convolutional  
Neural  
Networks

# Haar Cascade Classifier



- Haar Classifier е најактуелниот и најмоќниот метод за класификација и наоѓање на фаца/лице во слика

# Convolutional Neural Networks



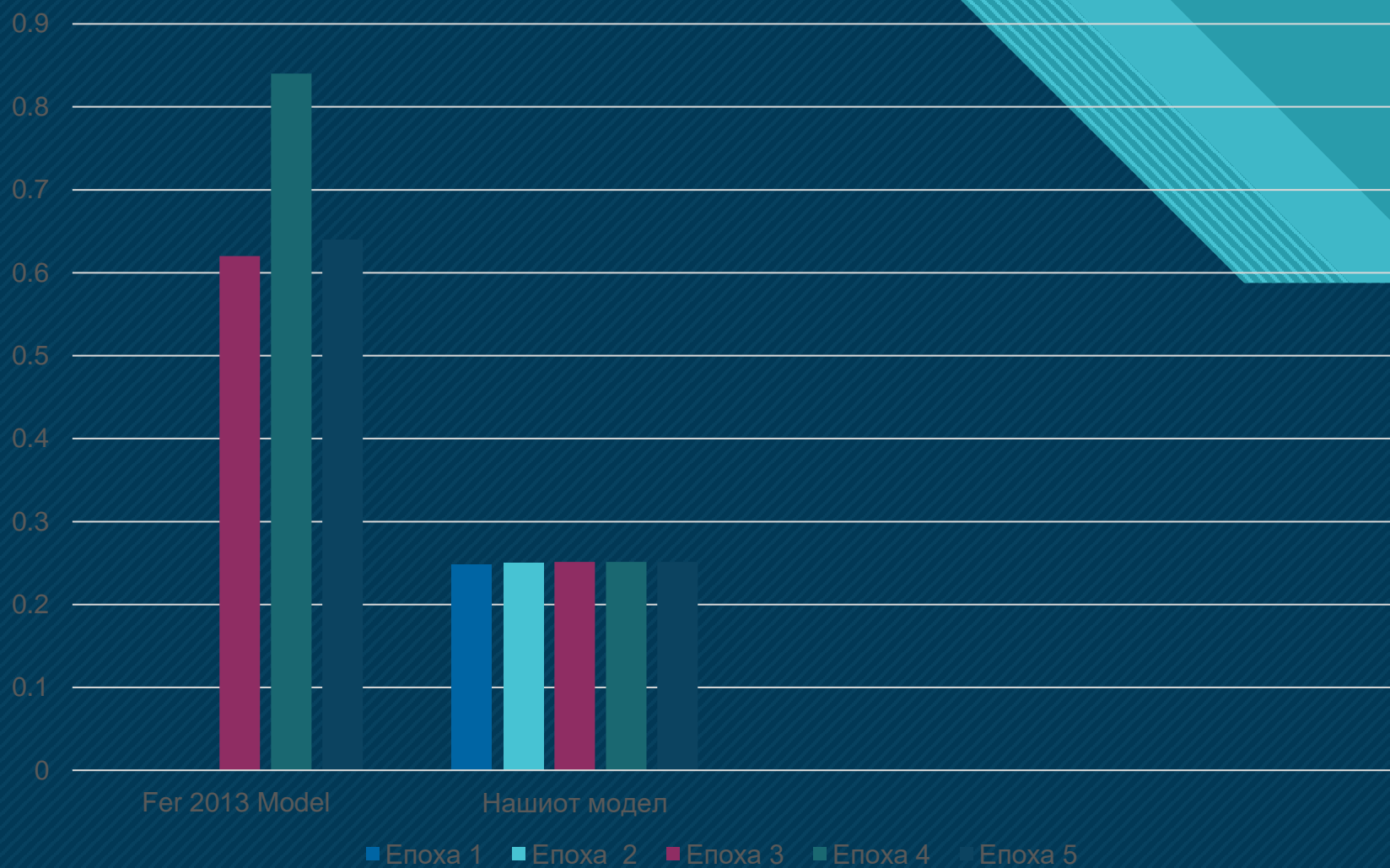
- Convolutional Neural Networks во моментот се најакутелните и најдобрите машинско учење алгоритми кога имаме потреба да обработуваме слики



# Результати



## Прецизност на моделите



Иако нашиот модел може успешно да предвиде емоции, сепак треба дополнителна работа и компјутерска моќ да се доведе до нивото на Fer 2013 моделот



# Ви благодарам за вниманието

Изработил  
Леонид Трајкоски 211169