

# IDENTIFYING LEAD CAST IN MOVIES USING NEURAL NETWORK MODEL

MINOR PROJECT REPORT

7<sup>TH</sup> SEMESTER

BACHELOR OF TECHNOLOGY

IN

COMPUTER ENGINEERING



UNDER THE SUPERVISION OF:

DR. TANVIR AHMAD

SUBMITTED BY:

RASHID AZIZ (15BCS0047)

MD. MUSHARRAF (15BCS0044)

MD. HAMID (15BCS0045)

DEPARTMENT OF COMPUTER ENGINEERING  
FACULTY OF ENGINEERING AND TECHNOLOGY

JAMIA MILLIA ISLAMIA

NEW DELHI-110025

## **CERTIFICATE**

This is to certify that the project entitled “**Identifying lead cast in movies using neural network model**” by **Rashid Aziz (15BCS0047)**, **Md. Musharraf (15BCS0044)** and **Md. Hamid (15BCS0045)** is a record of bonafide work carried out by them, in the Department of Computer Engineering, Jamia Millia Islamia under my supervision and guidance for Minor Project in seventh semester of Bachelor Of Technology in Computer Engineering, Jamia Millia Islamia in the academic year 2018.

**Prof. Tanvir Ahmad**

(Head of the Department)

Department of Computer Engineering

Faculty of Engineering and Technology

Jamia Millia Islamia

# ACKNOWLEDGEMENT

A very sincere and honest acknowledgement to **Mr. Tanvir Ahmad, Professor, Head of the Department**, Department of Computer Engineering, Jamia Millia Islamia, New Delhi for his invaluable technical guidance and support, great innovative ideas and overwhelming support. We are also expressing our gratitude to the Department of Computer Engineering and entire faculty members, for their teaching, guidance and encouragement.

We are also thankful to our classmates and friends for their valuable suggestions and support whenever required.

We regret any inadvertent omissions.

**Md. Musharraf**

(15BCS0044)

B. Tech (7<sup>th</sup> Semester)

**Rashid Aziz**

(15BCS0047)

B. Tech (7<sup>th</sup> Semester)

**Md. Hamid**

(15BCS0045)

B. Tech (7<sup>th</sup> Semester)

Department of Computer Engineering

Faculty of Engineering & Technology

JAMIA MILLIA ISLAMIA, NEW DELHI

# CONTENT

1. Introduction .....	5
1.1 Abstract .....	6
1.2 Need for facial recognition .....	6
1.3 Challenges .....	7
1.4 Transfer Learning .....	7
1.5 Tools used .....	9
1.5.1 Deep Learning .....	9
1.5.2 Image Processing .....	9
1.6 Network Architecture .....	9
2. Literature survey .....	13
2.1 Abstract .....	13
2.1.1 Labeled Faces in the Wild .....	13
3. Implementation .....	15
3.1 Proposed Framework .....	15
3.2 Flow Chart .....	17
4. Experimental results .....	18
4.1 Results and Accuracy Metrics .....	18
4.2 Screenshots .....	18
5. Conclusion and Future work .....	20
6. References .....	21

# 1. INTRODUCTION

## 1.1 Abstract

Detecting and naming actors/actress in movies are important for content based indexing and retrieval of movie scenes and can also be used to support statistical analysis of the film style. Detecting and naming actors/actress in unedited footage can be useful for post-production.

Recognizing human faces in the wild is emerging as a critically important and technically challenging computer vision problem/image processing problem. With a few notable exceptions, most previous works in the last several decades have focused on recognizing faces obtained in a laboratory setting. However, with the introduction of databases, face recognition community is gradually shifting its focus on much more challenging and unconstrained settings. Hence, to further boost the unconstrained face recognition research, this project proposes a methodology that has much more variability compared to previous approaches. Proposed methodology consists of collecting faces of many known actors collected from their known movies. In contrast to the other mechanisms proposed in the literature, which used face detectors to automatically detect the faces from the web collection, images in this proposed method are generated manually. Manual selection of faces from movies may result in high degree of variability (in scale, pose, expression, illumination, age, occlusion, makeup) which one could ever see in natural world. This method include three main iterations:

Preprocessing, Feature Extraction and Classification. This Study has been carried out using facial images of age 18-60 years consisting of both gender types. Classification is done using Convolutional Neural Network. Proposed method will provide a detailed annotation in terms of age, pose, gender, expression, amount of occlusion, for each face, which may help other face related applications.

## **1.2 Need for Facial Recognition:**

Face recognition is the task of identifying an already detected object as a known or unknown face, and in more advanced cases, telling exactly whose face it is. Face recognition is an easy task for humans but not so easy for computers. While initially it was a form of computer application, it has seen wider uses in recent times on mobile platforms and in other forms of technology, such as robotics. It is typically used as an access control in security systems (i.e access through facial recognition) and can be compared to other biometrics such as fingerprint or eye iris recognition systems. Although the accuracy of facial recognition system as a biometric security system technology is lower than iris recognition and fingerprint recognition, it is widely adopted due to its contactless and non-invasive process. Recently, it has also become popular as a commercial identification and marketing tool as well as for entertainment section too. Other applications include advanced human-computer interaction, video surveillance, automatic indexing of images, and video database, Chinese Social Credit System (proposed), Dubai Happiness rating system (proposed).

### **1.3 Challenges:**

- Illumination
- Expression
- Background
- Pose
- Variation
- Complexity

### **1.4 Transfer Learning**

Transfer learning is a research problem ,in machine learning, that focuses on storing knowledge gained through solving one problem and applying it to a different but a bit related problem. For example, the knowledge gained while learning to recognize trucks could apply when trying to recognize cars.

Machine learning algorithms are typically designed to address the isolated tasks. Through transfer learning, methods are developed to transfer the gathered knowledge from one or more of these source tasks to improve learning in a related target task. The goal of this transfer of learning strategies is help to evolve machine learning to make it as efficient as human.

During transfer learning, knowledge is leveraged from a source task to improve learning in a related totally new task. If the transfer method ends up decreasing performance on the new task, it is called a

negative transfer. A major challenge when developing transfer methods is ensuring positive transfer between all the related tasks while still avoiding negative transfer between less or not related tasks.

When applying gathered knowledge from one task to the another, the original task's characteristics are usually mapped onto those of the other's to specify correspondences.

Transfer learning is also useful during deployment of upgraded technology such as a Chabot. If the new task is similar enough to previous task, transfer learning can assess which knowledge should be transplanted into the next. Using transfer learning, developers can decide which knowledge and data is reusable from the previous task, and transfer that information for use when developing the upgraded version.

The effectiveness of transfer learning techniques is measured using three common indicators: One is measuring whether performing the target task is achievable using only the transferred knowledge. Second is measuring the amount of time it takes to learn the target task using knowledge gained from transferred learning versus how long it would take to learn without it.

Third is whether the final performance of the task learned via transfer learning is comparable to completion of the original task without the transfer of knowledge to the target task. Many deep neural networks trained on natural images exhibit an interesting common phenomenon: on the first layer they learn features similar to Gabor filters and color blobs. Such first-layer features appear not to specific to a particular dataset or task but are general as they are applicable to many datasets and tasks. As finding these standard features on the first layer seems to occur totally regardless of the exact cost function and natural image dataset and we call these first-layer features general. For example, in a network with an N-dimensional softmax output layer that has been successfully trained towards a supervised classification objective, each output unit will be specific to a particular class. We thus call the last-layer features specific.



In transfer learning we first train a base network on a base dataset and task, and then we reroute the learned features, or transfer them, to a second target network to be trained on a target dataset/task. This process will tend to work if the features are general, that is, suitable to both original tasks and target tasks, instead of being specific to the original task.

## **1.5 Tools Used:**

### **1.5.1 Deep Learning:**

- Tensorflow
- Keras
- Scikit Learn

### **1.5.2 Image Processing:**

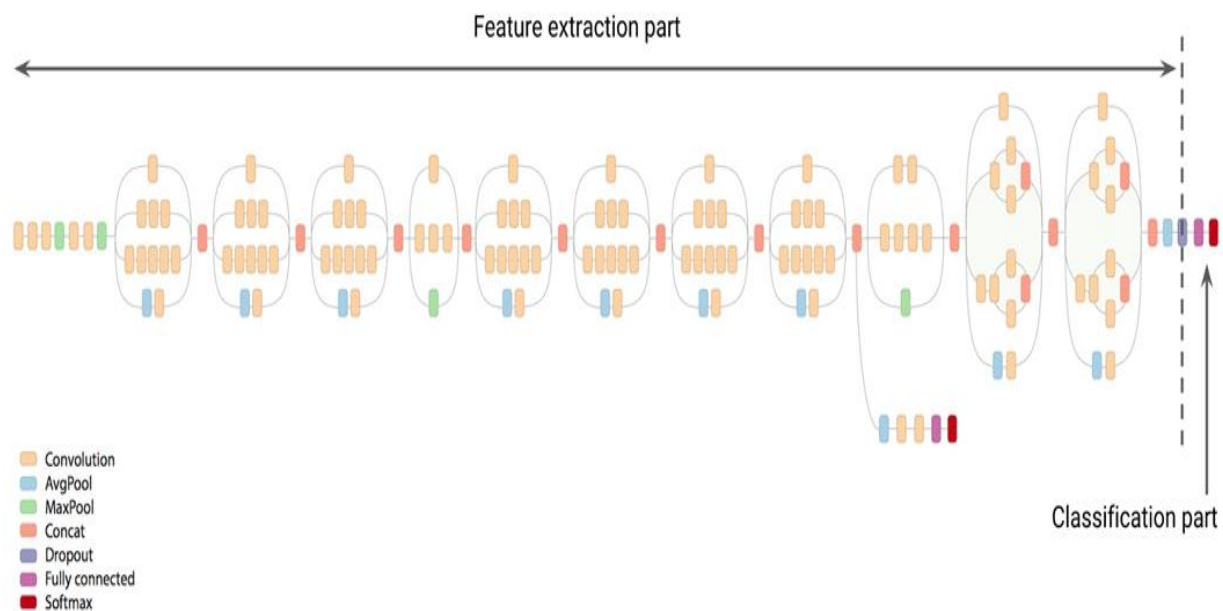
- Opencv
- Haar-Cascade Classifier

## **1.6 Network Architecture:**

In practice, very few people train an entire Convolutional Network from scratch because it is relatively tough to have a dataset of sufficient size. Instead, it is very common to pre-train a Convolutional Network on a very large dataset (e.g. ImageNet, which contains 1.2 million images with 1000 categories), and then use that Convolutional Network either as an initialization or a fixed feature extractor for the task of interest.

Inception was developed at Google to provide the state of the art performance on the ImageNet, Large-Scale Visual Recognition Challenge and to be more computationally efficient than its competitor architectures as well as having more accuracy. However, what makes Inception exciting is that its architecture can be applied to a whole host of other learning problems in computer vision without retraining again and again, with a better accuracy

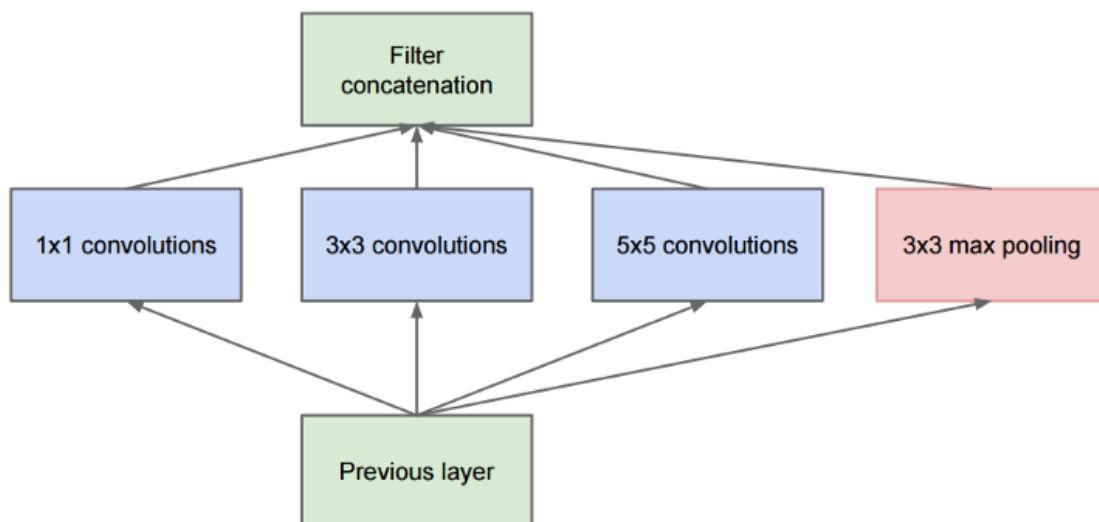
### High level Overview of Inception Model



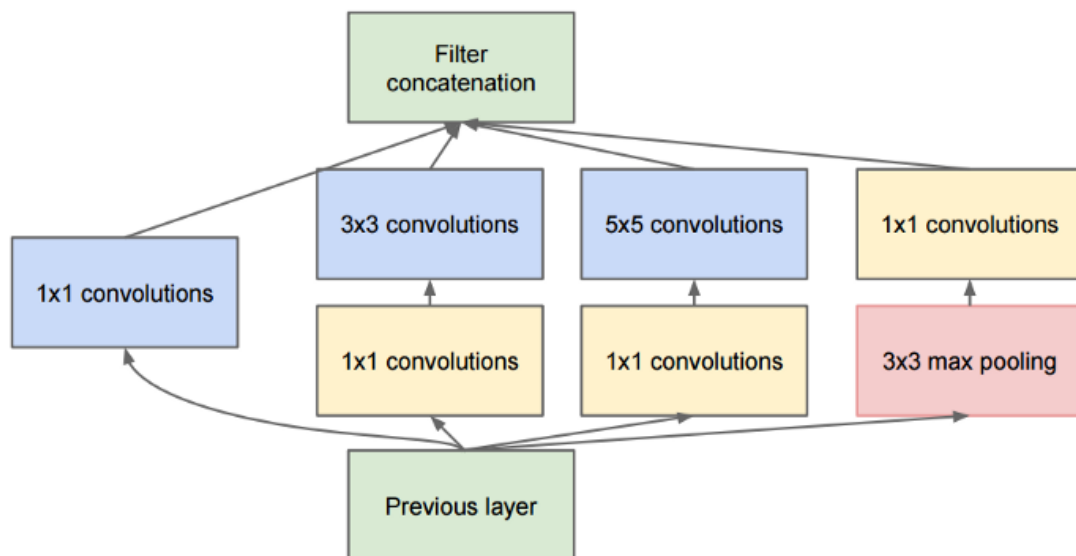
Inception is:

- An  $299 \times 299 \times 3$  input representing a visual field of 299 pixels and 3 color (RGB) channels
- Five vanilla convolution layers, with some interspersed max-pooling operations.
- Successive stacks of “Inception Modules”
- A softmax output layer at the end (logits) and at an intermediate output layer (aux\_logits) just after the mixed  $17 \times 17 \times 768$  layer.

While stacking Inception modules leads to depth, each module is also “wide” and architected to recognize features at multiple length scales. That means introducing convolutions with several filter sizes; in particular in Inception, that means including  $3 \times 3$  and  $5 \times 5$  convolutions in each stacked module:



The downside is that these convolutions are expensive, especially when repeatedly stacked in a deep learning architecture. To combat these problems, Inception's architects stacked 1x1 convolutions in front of the expensive 3x3 and 5x5 convolutions to reduce the dimensionality before each convolution:



## 2. LITERATURE SURVEY

### **2.1 Abstract:**

As one of the most successful applications of image analysis and identification, face recognition has recently received significant attention, especially during the past several years. Two reasons accounting for these trend: the first is the wide range of commercial and law enforcement applications, and the second is the availability of feasible technologies after 30 years of research. Even though current machine recognition systems have reached a certain level of maturity and accuracy yet their success is limited by the conditions imposed by many real applications. For example, recognition of face images collected in an outdoor environment with variations in illumination and/or pose remains a largely tough and unsolved problem. In other words, current image identification systems are still far away from the capability of the human perception system.

### **2.2 Labeled Faces in the Wild:**

In 2007, Labeled Faces in the Wild (LFW) was released in an effort to spur research in face recognition, specifically for the problem of face verification with unconstrained images. Since that time, more than 50 papers have been published to improve upon this benchmark some successful and some not so successful. There were several aims behind the introduction of LFW. These included

- Stimulating research on face recognition in unconstrained/Wild images
- Providing an easy-to-use database, with a low barriers to the entry, easy browsing, and multiple parallel versions to lower pre-processing burdens.
- providing consistent and precise protocols for the use of the database to encourage fair and meaningful comparisons
- Collecting the results to allow easy comparison, and easy replication of results in new research papers.

# 3. IMPLEMENTATION

## **3.1 Proposed Framework:**

Our method comes with collecting detailed annotation in terms of age, bounding box, movie release, expression, gender, pose, makeup. The database is designed through following steps:

- 1) Selection of movies and actors,
- 2) Selection of frames from videos,
- 3) Cropping of faces,
- 4) Pruning the database

### **Selection of movies and actors:**

Identification of actors and movies became the critical part in designing the database and optimizing the human labor. First, in order to ensure the diversity in appearance, we will select the movies from wide range. All the movies will be collected from personal collection and YouTube. In the second step, we will select the actors that have a long career span so that we can obtain multiple movies of the actors. For each actor, we will select the movies that give wide variations in age. As far as

resolution and quality is concerned, old films were at poorer resolution while new one at available at different resolutions. This will result in variation in terms of resolution and quality of images. The number of movies selected for each actor will be varied from 2 – 5. Since the images will be extracted through a manual process, it is important to minimize the number of movies as much as possible in order to reduce the manual labor. During this stage, we will carefully select the movies in such a way that there is a maximum overlap of actors across movies.

### **Selection of frames:**

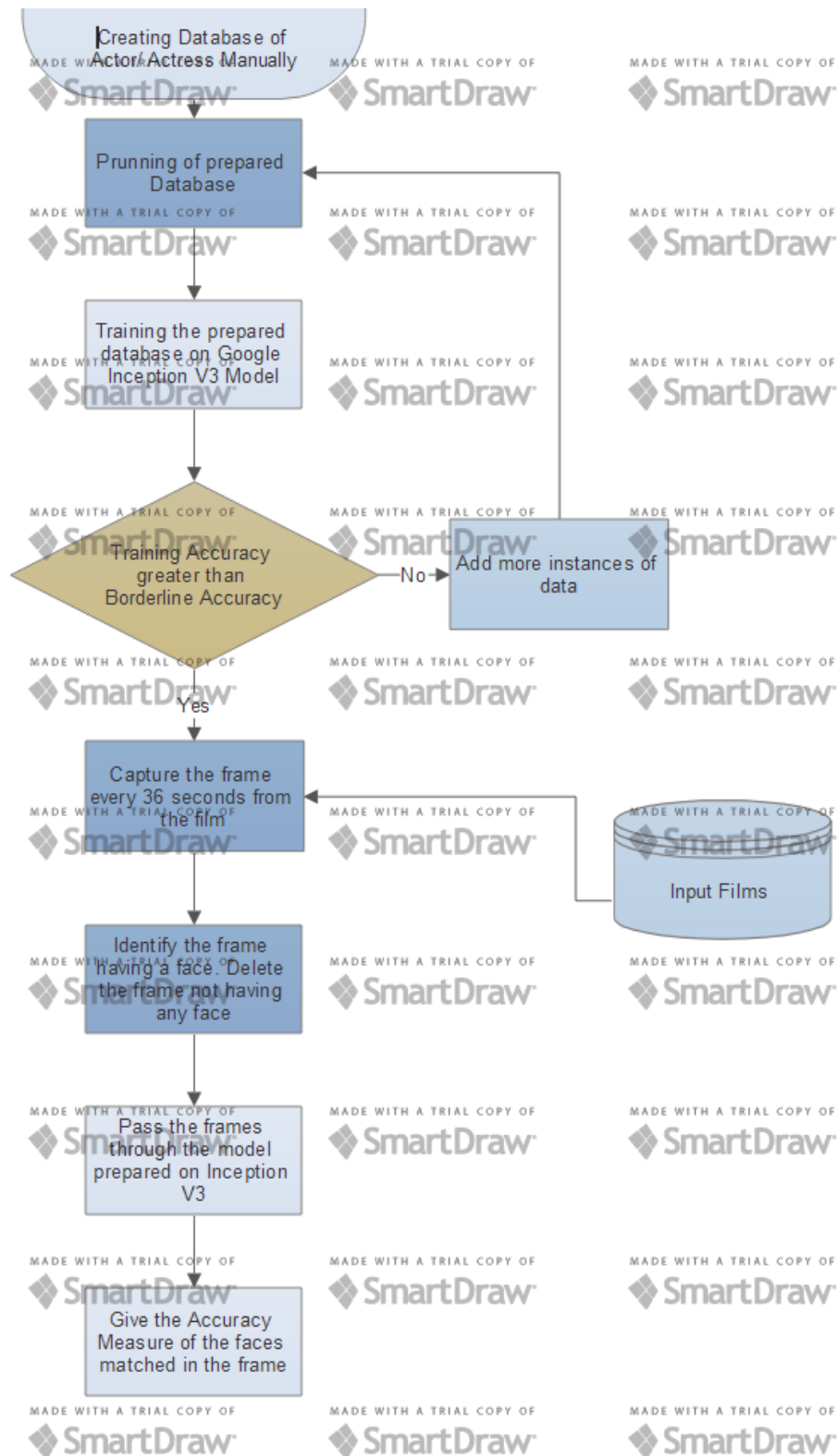
Once the movies are collected, we are going to extract the frames from these videos with frame interval of 36 seconds. We will prune the database based on certain condition: First, only one frame with signification variation from a shot unless there is another frame with significant difference with the first frame. Second, if there are multiple variations available in a single shot, faces with better angle and pose variation, which offer a serious challenge to recognition algorithms compared to facial expression and illumination, are preferred. Third, those frames will not be considered with small faces and difficult to recognize manually.

**Pruning the database:** As a post-processing step, through a careful inspection removing of any duplicates or similar images for each subject will be done.

**Feeding into Neural Network model:** Finally we will feed our database into our neural network by dividing our dataset in 30-70 ratio. Model will train on 70% and rest 30% will be used for testing.



## 3.2 Flow Chart:



# 4. Experimental Results

## 4.1 Results and Accuracy Metrics:

Input Film: Avengers Civil War

Correct Leading Cast:

Male: Robert Downey Jr., Chris Evans, Don Cheadle

Female: Scarlett Johansson, Elizabeth Oslen

Output:

Training Accuracy: **85.0%**

Validation Accuracy: **75.0%**

Final test Accuracy: **75.7%**

Predicted Leading Cast:

Male: Don Cheadle

Female: Elizabeth Oslen

## 4.2 Screenshots:











```
Leading male cast: don cheadle : 24.651412508548145
Leading female cast: elizabeth oslen : 60.553036475287335
>>> |
```



```
/usr/bin/bash --login -i E:\projects\minor\codes\final codes\train.sh
2018-12-21 15:47:32.465744: Step 440: Train accuracy = 85.0%
2018-12-21 15:47:32.465744: Step 440: Cross entropy = 0.925749
2018-12-21 15:47:32.651865: Step 440: Validation accuracy = 74.0% (N=100)
2018-12-21 15:47:34.521419: Step 450: Train accuracy = 87.0%
2018-12-21 15:47:34.521419: Step 450: Cross entropy = 0.874617
2018-12-21 15:47:34.704539: Step 450: Validation accuracy = 71.0% (N=100)
2018-12-21 15:47:36.567548: Step 460: Train accuracy = 89.0%
2018-12-21 15:47:36.568550: Step 460: Cross entropy = 0.812116
2018-12-21 15:47:36.751672: Step 460: Validation accuracy = 64.0% (N=100)
2018-12-21 15:47:38.617296: Step 470: Train accuracy = 90.0%
2018-12-21 15:47:38.617296: Step 470: Cross entropy = 0.744548
2018-12-21 15:47:38.799417: Step 470: Validation accuracy = 71.0% (N=100)
2018-12-21 15:47:40.674262: Step 480: Train accuracy = 82.0%
2018-12-21 15:47:40.674262: Step 480: Cross entropy = 0.893730
2018-12-21 15:47:40.855382: Step 480: Validation accuracy = 77.0% (N=100)
2018-12-21 15:47:42.722537: Step 490: Train accuracy = 86.0%
2018-12-21 15:47:42.722537: Step 490: Cross entropy = 0.825582
2018-12-21 15:47:42.906790: Step 490: Validation accuracy = 72.0% (N=100)
2018-12-21 15:47:44.596317: Step 499: Train accuracy = 85.0%
2018-12-21 15:47:44.596317: Step 499: Cross entropy = 0.775020
2018-12-21 15:47:44.778438: Step 499: Validation accuracy = 75.0% (N=100)
Final test accuracy = 75.7% (N=206)
Converted 2 variables to const ops.
Training finished
```

## Original Cast

Cast (in credits order) complete, awaiting verification

	Chris Evans	...	Steve Rogers / Captain America
	Robert Downey Jr.	...	Tony Stark / Iron Man
	Scarlett Johansson	...	Natasha Romanoff / Black Widow
	Sebastian Stan	...	Bucky Barnes / Winter Soldier
	Anthony Mackie	...	Sam Wilson / Falcon
	Don Cheadle	...	Lieutenant James Rhodes / War Machine
	Jeremy Renner	...	Clint Barton / Hawkeye
	Chadwick Boseman	...	T'Challa / Black Panther
	Paul Bettany	...	Vision
	Elizabeth Olsen	...	Wanda Maximoff / Scarlet Witch

# 5. CONCLUSION AND FUTURE WORK

In this project, we have introduced a new method for face recognition based on transfer learning using Google ImageNet Inception V3 model. It is developed with the intention of providing a common benchmark for face recognition and identification.

The main characteristics are:

- 1) Large diversity in terms of pose, age, expression, illumination, make-up and the combined variations.
- 2) Selection of frames and bounding box based on presence of faces in it.

By making this project available to the research community, we hope to encourage the exploration of many unsolved problems.

# REFERENCES

1. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens “Rethinking the Inception Architecture for Computer Vision”
2. A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, “From few to many: Illumination cone models for face recognition under variable lighting and pose,” in PAMI, 2001.
3. A. Mart´ınez and R. Benavente, “The AR face database,” in CVC Technical Report, 1998.
4. T. Sim, S. Baker, and M. Bsat, “The CMU Pose, Illumination, and Expression Database,” in PAMI, 2003.
5. P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, “Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection,” in PAMI, 1997.
6. J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust Face Recognition via Sparse Representation,” in PAMI, 2009.
7. G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, “Labeled faces in the wild: A database for studying face recognition in unconstrained environments,” University of Massachusetts, Amherst, Tech. Rep., 2007.
8. N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, “Attribute and simile classifiers for face verification,” in ICCV, 2009.
9. C. Klontz and A. K. Jain, “A Case Study on Unconstrained Facial Recognition using the Boston Marathon Bombings suspects, Michigan State University, Tech. Rep., 2013.