End Sem Review

An Intelligent Pose Recommendation System using feature fusion for Humans

Students:
111807032 - Sakshi Patil
111807039 - Piyusha Taware
111807053 - Sonali Sargar

Mentors:
Prof. Dr. Abhishek Bhatt
Mr. Suresh Gara

Table of Contents

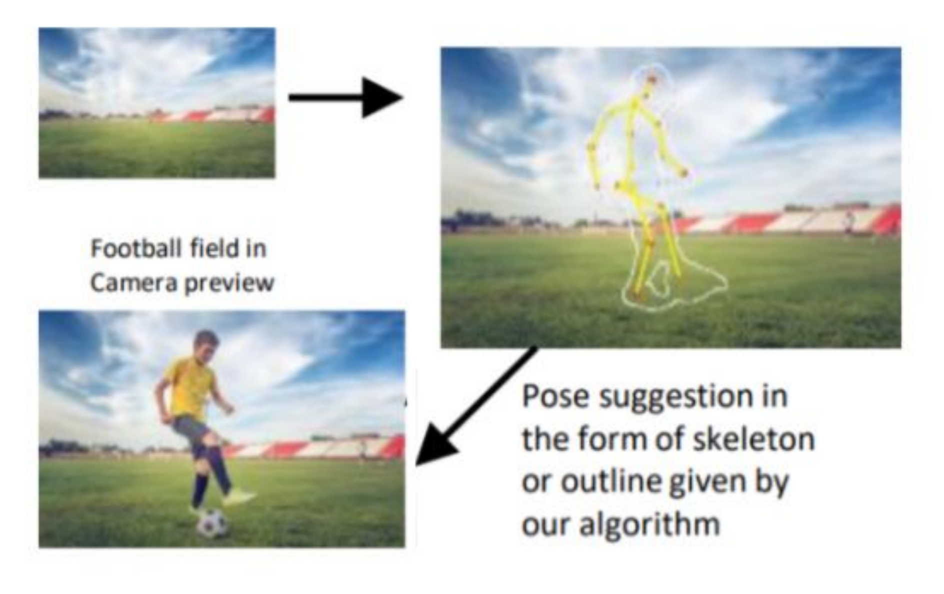
Introduction

Increase in the number of users on social media

Most of the pictures taken end up in social media.

Lack of professionalism in the pictures and degradation of quality.

Proposes intelligent photo pose recommendation method.



Problem statement

Design and develop a Pose recommendation model to recommend a better pose. Build a deep learning based model that is trained on a huge database covering different location, background and foreground.

Project Objectives

Goal 1

To study and implement previous available techniques for pose estimation and recommendation.

Goal 2

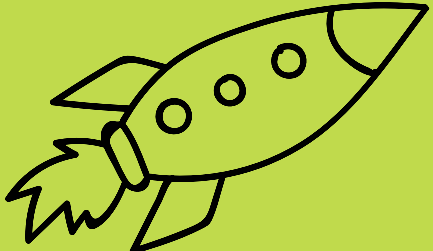
To study and implement various models useful for feature extraction and fusion.

Goal 3

To develop a robust learning based solution/ algorithm to predict a pose.

Goal 4

Accuracy assessment and testing the implementation.



Human Pose Estimation

- Human Pose Estimation: It determines the human posture and identifies the body's key points in pictures and videos.

- The human pose estimation techniques can be broadly divided as single person and multi person pose estimation approaches.

- In Single person approach, the objective is to find the keypoint position in desired area, whereas in multiperson approach the objective is to solve an unconstrained problem as the number and position of persons are unknown.

- In Single person estimation is further divided into direct regression-based and heatmap based approaches.

Human Pose Estimation

- **Multi-person approaches can be classified as Top-down approach and Bottom-up approach.**

- **The top-down approach identifies and localizes individual person instances by bounding box object detector which is followed by estimation of body joints.**

- **The Bottom-up approach starts by estimating each body joint first and then grouping them to form a unique pose.**

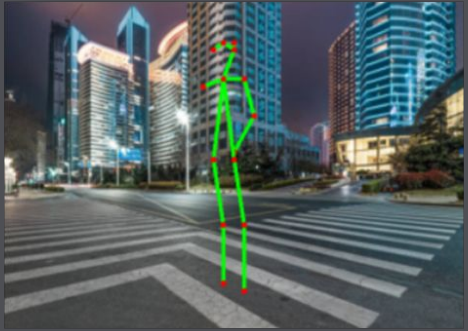
Human Pose Recommendation

- It is a methodology to enhance quality of photography of naive users.
- It aims at recommending the best possible human pose to click professional level photographs, based on the background of a given image.

Example:

Query Image

Recommended Pose



Literature Review

So, the most popular human pose estimation methods are :

1) **Alpha Pose** : From bounding box an individual's region is extracted. Now, in this extracted region, a single person pose estimator is used to determine the human pose skeleton. The approximated individual pose is remapped back to the original picture coordinate system. Lastly, redundant pose deductions are solved using non-maximum suppression techniques.

2) **Deep Cut** : It first spots the feasible body parts which are jointly clustered and then labeled separately as a leg, arm, etc. The next task is to separate the body parts of each person. Then we completely group all the observed key points in the given input, and the output that occurs will coincide with the skeleton representation of the human body.

Literature Review

3) Deeper Cut : The process begins with first randomly selecting a single person in the photo. Then we fix the position for every keypoint and then predict the location of a particular body part. Then, we evaluate pairwise probability for every location in the image. Finally, the valid human pose is estimated.

4) Convolutional Pose Machine(CPM) : CPM has a multiple stages architecture, and at each step the belief map is created, it helps in the identification of keypoints. At the first stage the number of joints of the individual in the image are identified. Also the number of layers in the belief map are identified.

Literature Review

5) **Iterative Error Feedback :** IEF works on the mechanism of prediction, later on it is about identifying what is incorrect in the prediction and then correcting them. In this self-correcting model is used.

6) **Stacked Hourglass Network :** In the Stacked hourglass architecture, the network comprises successive steps of pooling and sampling layers. The network collects pieces of information such as an individual's posture and limb articulation at each scale of the video or RGB image. Hourglass network outputs pixel-wise predictions as it gathers all these features accurately.

Literature Review

7) HRNet: It begins with a high-resolution subnetwork as the first stage, high-to-low resolution subnetworks are gradually added one after another in parallel to form more stages. Multi-scale fusions are conducted repeatedly for information exchange across subnetworks

8) Open Pose: It first determines the location of each joint in input image, then the orientation of the body parts. The Biporate matching is performed on these body part candidates and finally the pose is estimated for every person.

9) Dark Pose: In Dark Pose, the heatmaps are predicted which are further decoded into final joint coordinates. It allows the use of small resolution input images.

Pose Estimation

Model	Architecture	Single/Multi Person	Approach(TD/BU)	Dataset Used	Evaluation Metrics
Alpha Pose	VGG	Multi	TD	MPII, COCO	AP, mAP, PCKh@0.5
Deepcut	VGGNet	Multi	BU	MPII, LSP, WAF	PCKh@0.5, PCK, mPCP, mAP
Deeper Cut	ResNet	Both Single and Multi	BU	COCO, LSP, MPII	AP, mAP, AUC, PCKh@0.5
CPM	VGG architecture	Single	TD	FLIC, LSP, MPII	PCK@0.1, PCK@0.2, PCK@0.5
IEF	VGG and ConvNet	Single	TD	LSP, MPII	PCKh@0.5

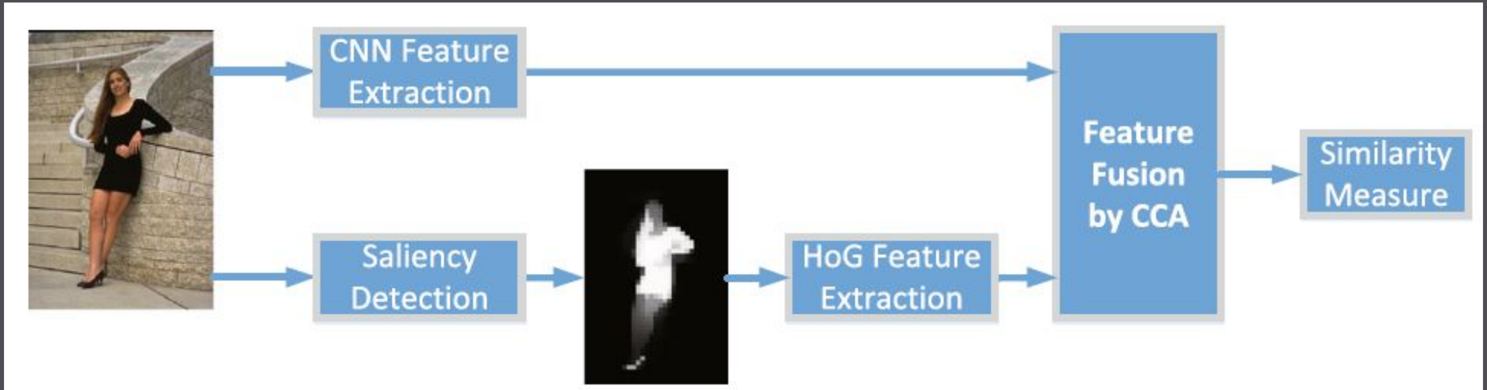
Pose Estimation

Model	Architecture	Single/Multi Person	Approach(TD/BU)	Dataset Used	Evaluation Metrics
Stacked Hourglass	ResNet	Single	Both TD and BU	FLIC, MPII, COCO	PCKh@0.5, PCK, PCK@0.2, AP, mAP
Open Pose	VGG	Multi	BU	COCO, MPII	AP, mAP, PCKh@0.5
HRNet	ResNet	Multi	BU	COCO, MPII	AP, mAP, PCKh@0.5
Dark Pose	ResNet, HRNet-W32	Single	TD	COCO, MPII	AP, PCK, OKS, PCKh@0.5

Literature Review

Adaptive recommendation for photo pose via deep learning [10]

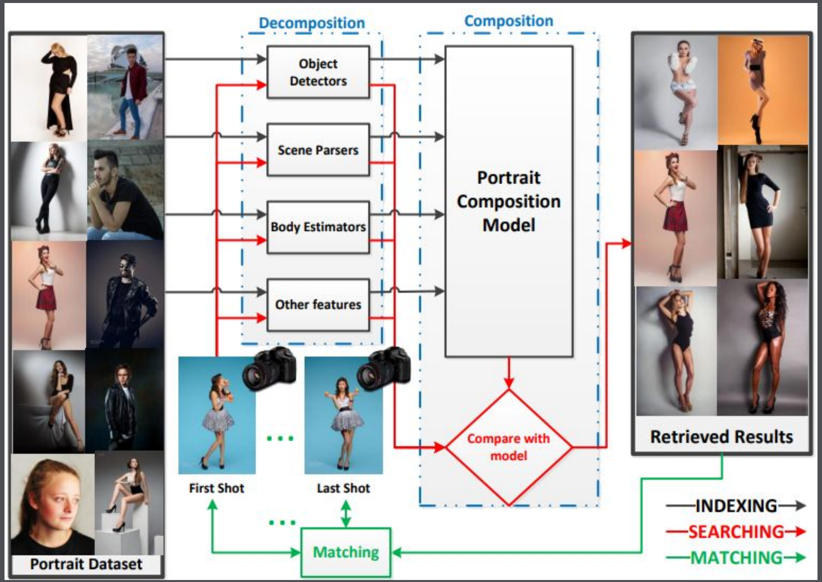
- This paper proposed fusion of global and local features for pose recommendation.
- VGG16 was used for extracting global features and HoG for local features.
- Salient Region Detection is used to find the Region Of Interest
- Euclidean distance is calculated for similarity metric.
- Their professional photo dataset included 50000 images of various backgrounds which they have collected from websites having professional photos such as Flickr, Weibo and Foursquare.



Literature Review

Intelligent Portrait Composition Assistance [14]

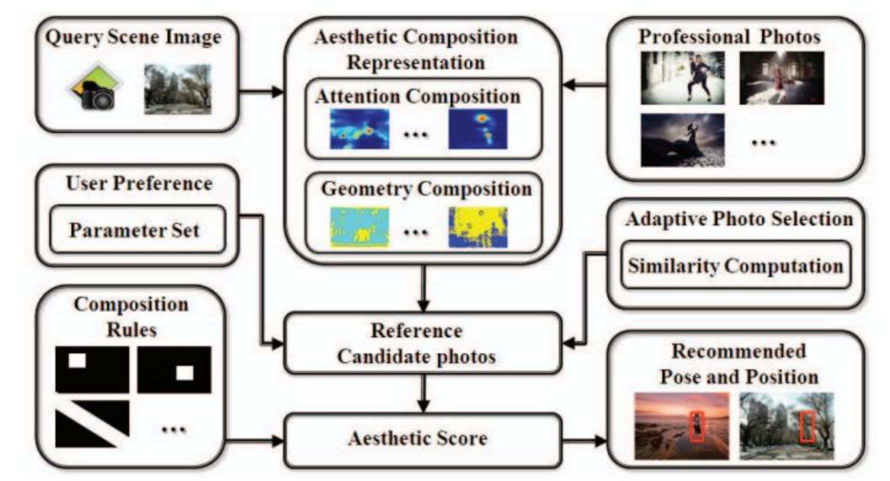
- It specifically addresses aesthetic retrieval and evaluation of the human poses in portrait photography, and provides meaningful and constructive feedback to photographers.
- The dataset consists of 320,000 images.
- This framework works by matching the semantics of query image and dataset of composed photos.
- This is done in order to optimize the human pose and other artistic aspects of a photo.



Literature Review

Aesthetic Composition Representation For Portrait Photographing [15]

- Their dataset consists of 232 images and 50 test images..
- Visual saliency model is used to extract the attention composition features.
- Spatial correlation is learnt with the geometry composition feature.
- Decaying exponential function used to weaken the magnitude of saliency while preserving spatial attention distribution.
- Hierarchical Kmeans method is used to find nearest-neighbors of the query images in the quantized 1024-dimension low level visual feature space



Approach 1

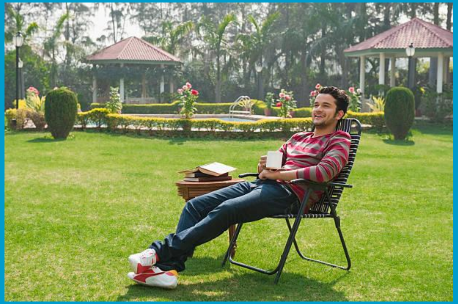
Pose Classification

1) Image Pose Classification

Dataset:

- We classified images based on the human pose.
- Wrote a python script for crawling photos from a professional photo website, namely StockSnap.
- Curated 185 images spanning 4 classes, namely: Sitting front pose, Sitting side pose, Standing front pose, Standing side pose

Sitting front pose



Sitting side pose

Standing front pose

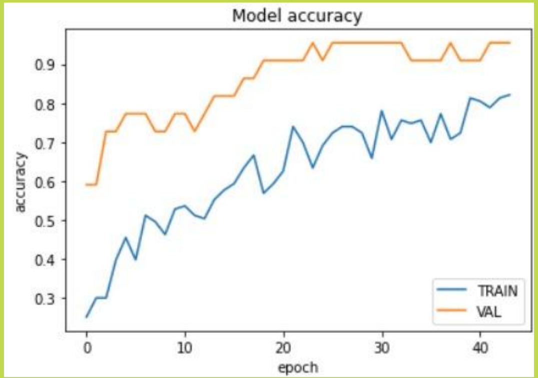
Standing side pose

1) Image Pose Classification

Pipeline for Pose Classification

Implementation: https://colab.research.google.com/github/tensorflow/tensorflow/blob/master/tensorflow/lite/g3doc/tutorials/pose_classification.ipynb#scrollTo=OsdqxGfxTE2HH

1) Image Pose Classification

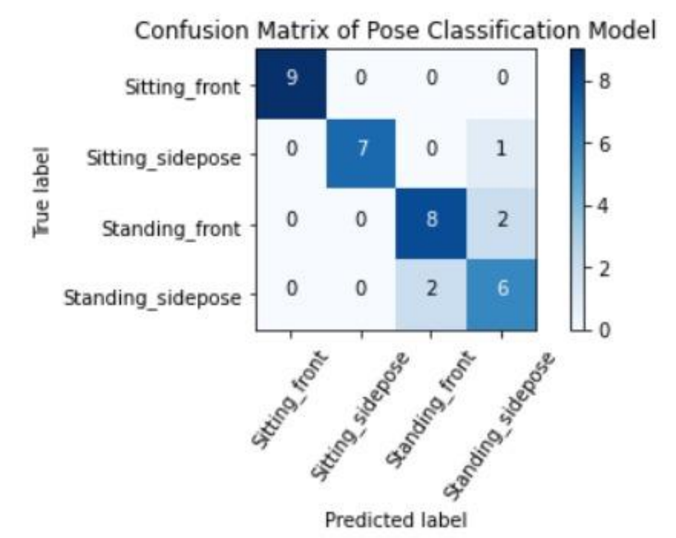


Output:

```
Epoch 44/200
1/8 [==>...........................] - ETA: 0s - loss: 0.4890 - accuracy: 0.8750
Epoch 44: val_accuracy did not improve from 0.95455
8/8 [==============================] - 0s 8ms/step - loss: 0.5161 - accuracy: 0.8211 - val_loss: 0.3787 - val_accuracy: 0.9545
```

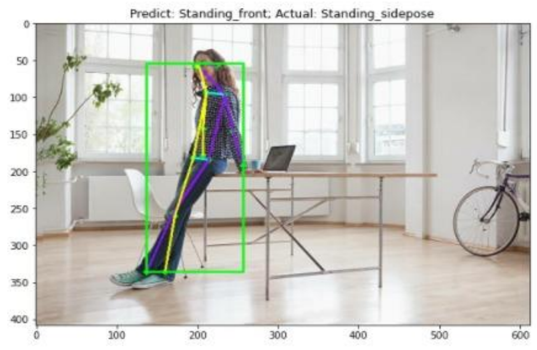
Classification Report:

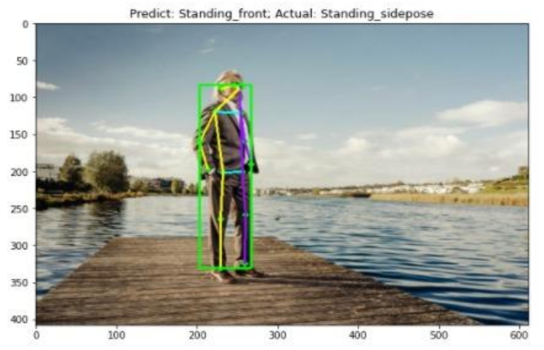
	precision	recall	f1-score	support
Sitting_front	1.00	1.00	1.00	9
Sitting_sidepose	1.00	0.88	0.93	8
Standing_front	0.80	0.80	0.80	10
Standing_sidepose	0.67	0.75	0.71	8
accuracy			0.86	35
macro avg	0.87	0.86	0.86	35
weighted avg	0.87	0.86	0.86	35



1) Image Pose Classification

Incorrect Classifications





Approach 2

Background Classification

2) Image Background Classification

Curated images spanning 9 different classes, classified them based on their background and human pose.

1. Snow

2. Glass door/ window

3. Wall

4. Greenery (Standing)

5. Greenery (Sitting)

6. Stairs

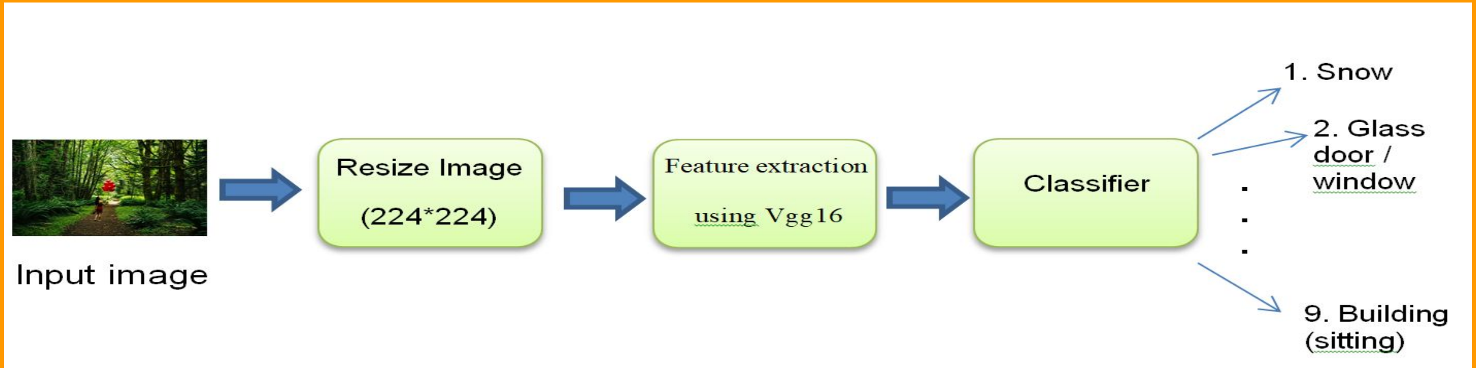
7. Beach

8. Building (standing)

9. Building (Sitting)

2) Image Background Classification

Pipeline for background classification:



Implementation:

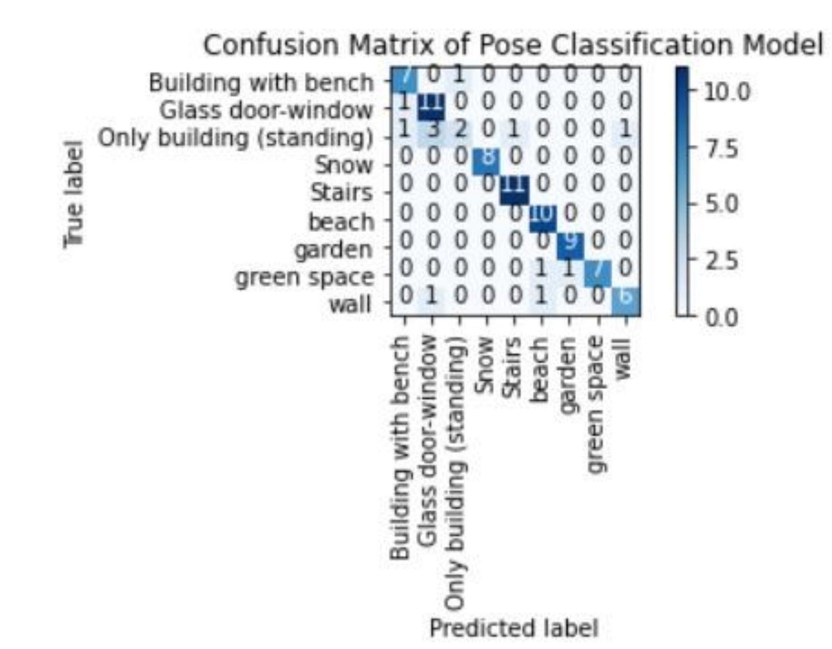
https://colab.research.google.com/drive/1Idf8shGgPdF0sfXx6Zs3-j81SkYFJI-I#scrollTo=9BEucVFnxrIInxrII

2) Image Background Classification

Output:

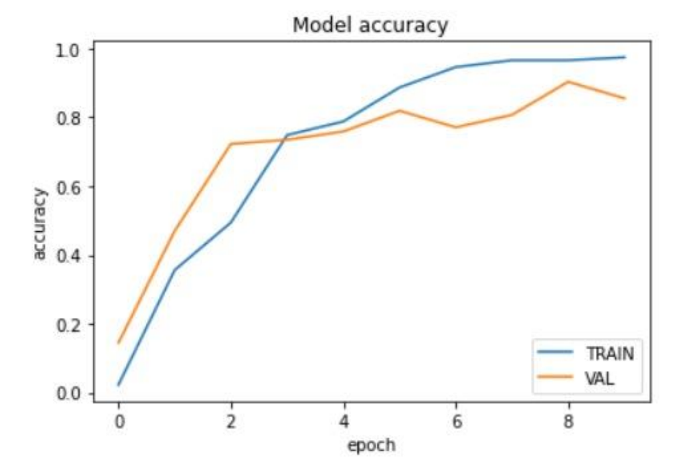
Classification Report

	precision	recall	f1-score	support
Building with bench	0.78	0.88	0.82	8
Glass door-window	0.73	0.92	0.81	12
Only building (standing)	0.67	0.25	0.36	8
Snow	1.00	1.00	1.00	8
Stairs	0.92	1.00	0.96	11
beach	0.83	1.00	0.91	10
garden	0.90	1.00	0.95	9
green space	1.00	0.78	0.88	9
wall	0.86	0.75	0.80	8
accuracy			0.86	83
macro avg	0.85	0.84	0.83	83
weighted avg	0.85	0.86	0.84	83



2) Image Background Classification

Output:




```
Epoch 3/10
12/12 [==============================] - 8s 677ms/step - loss: 1.4407 - accuracy: 0.4944 - val_loss: 0.8255 - val_accuracy: 0.7229
Epoch 4/10
12/12 [==============================] - 8s 685ms/step - loss: 0.7585 - accuracy: 0.7486 - val_loss: 0.8590 - val_accuracy: 0.7349
Epoch 5/10
12/12 [==============================] - 8s 693ms/step - loss: 0.7944 - accuracy: 0.7881 - val_loss: 0.7552 - val_accuracy: 0.7590
Epoch 6/10
12/12 [==============================] - 8s 683ms/step - loss: 0.3637 - accuracy: 0.8870 - val_loss: 0.5793 - val_accuracy: 0.8193
Epoch 7/10
12/12 [==============================] - 8s 690ms/step - loss: 0.2373 - accuracy: 0.9463 - val_loss: 0.5680 - val_accuracy: 0.7711
Epoch 8/10
12/12 [==============================] - 8s 682ms/step - loss: 0.1595 - accuracy: 0.9661 - val_loss: 0.5852 - val_accuracy: 0.8072
Epoch 9/10
12/12 [==============================] - 8s 693ms/step - loss: 0.1794 - accuracy: 0.9661 - val_loss: 0.4537 - val_accuracy: 0.9036
Epoch 10/10
12/12 [==============================] - 8s 745ms/step - loss: 0.1450 - accuracy: 0.9746 - val_loss: 0.4797 - val_accuracy: 0.8554
```

3) Unsupervised model - Dataset and similarity metrics

Dataset -

- Eliminated the use of classes to make dataset more generalised.
- Included images with more variety of background.
- Consists of 1067 images

Similarity Metrics -

- Cosine similarity

$$similarity = cos(\Theta) = S(I_q, I_{id}) = \frac{\sum_{i=1}^{N} f(I_q) . f(I_{id})}{\sqrt{\sum_{i=1}^{N} f(I_q)^2} \sqrt{\sum_{i=1}^{N} f(I_{id})^2}}$$

- Euclidean distance

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

- Chi - square distance

$$X^2 = \frac{1}{2} \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{(x_i + y_i)}$$

Local features

To extract local features of a given image we use color histogram in the following way:
For the image descriptor, we divide our image into five different regions as shown in figure 3:

1. the top-left corner,
2. the top-right corner,
3. the bottom-right corner,
4. the bottom-left corner,
5. the center of the image.

As shown in the image.

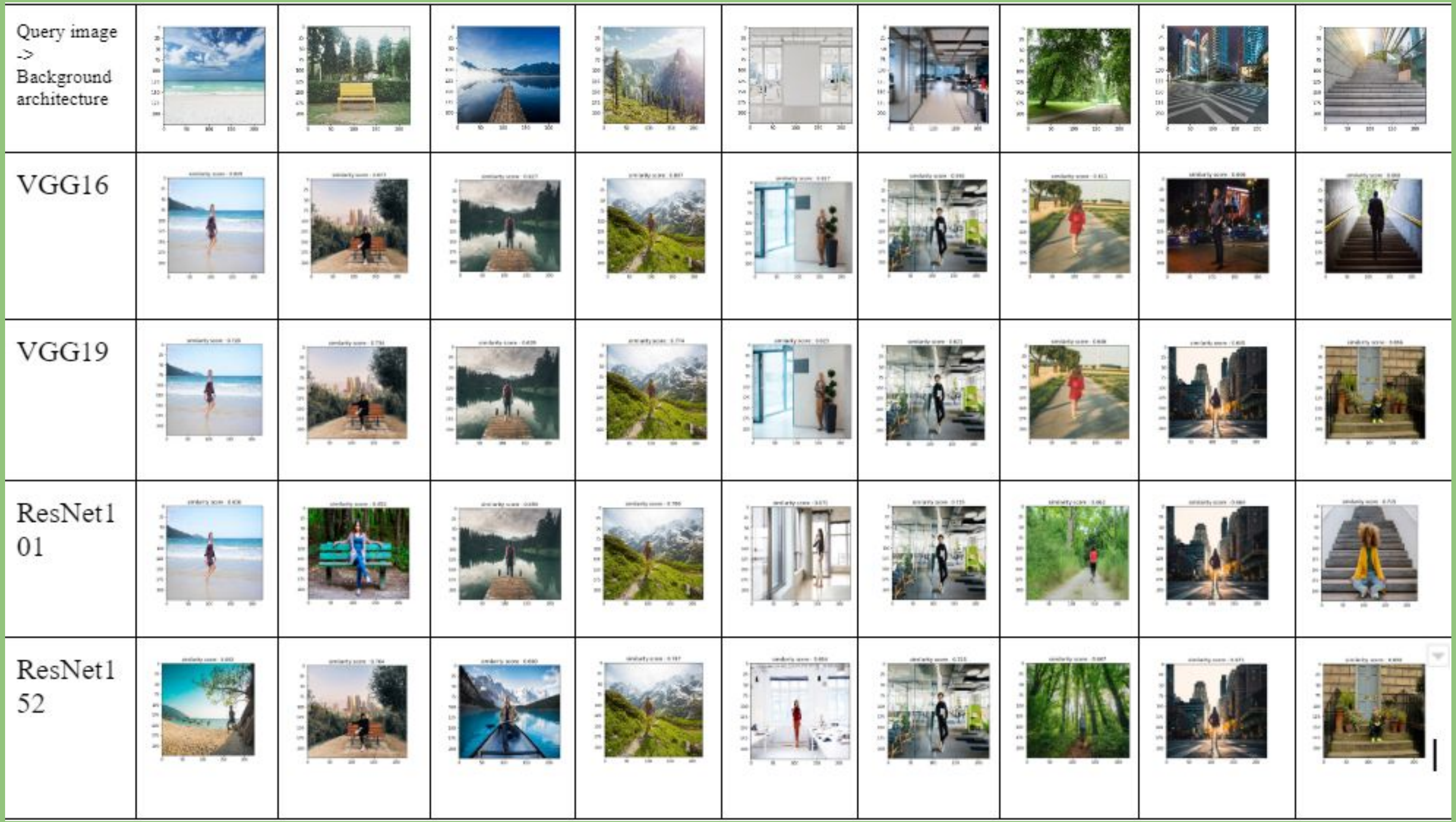
- Obtained 3D color histogram in the HSV color space with ratio 12:8:3.
- Feature vector of each region is of dimension 288(12 x 8 x 3).
- Dimension of local feature vector obtained is 1440(= 288 x 5)

Global features and Feature Fusion

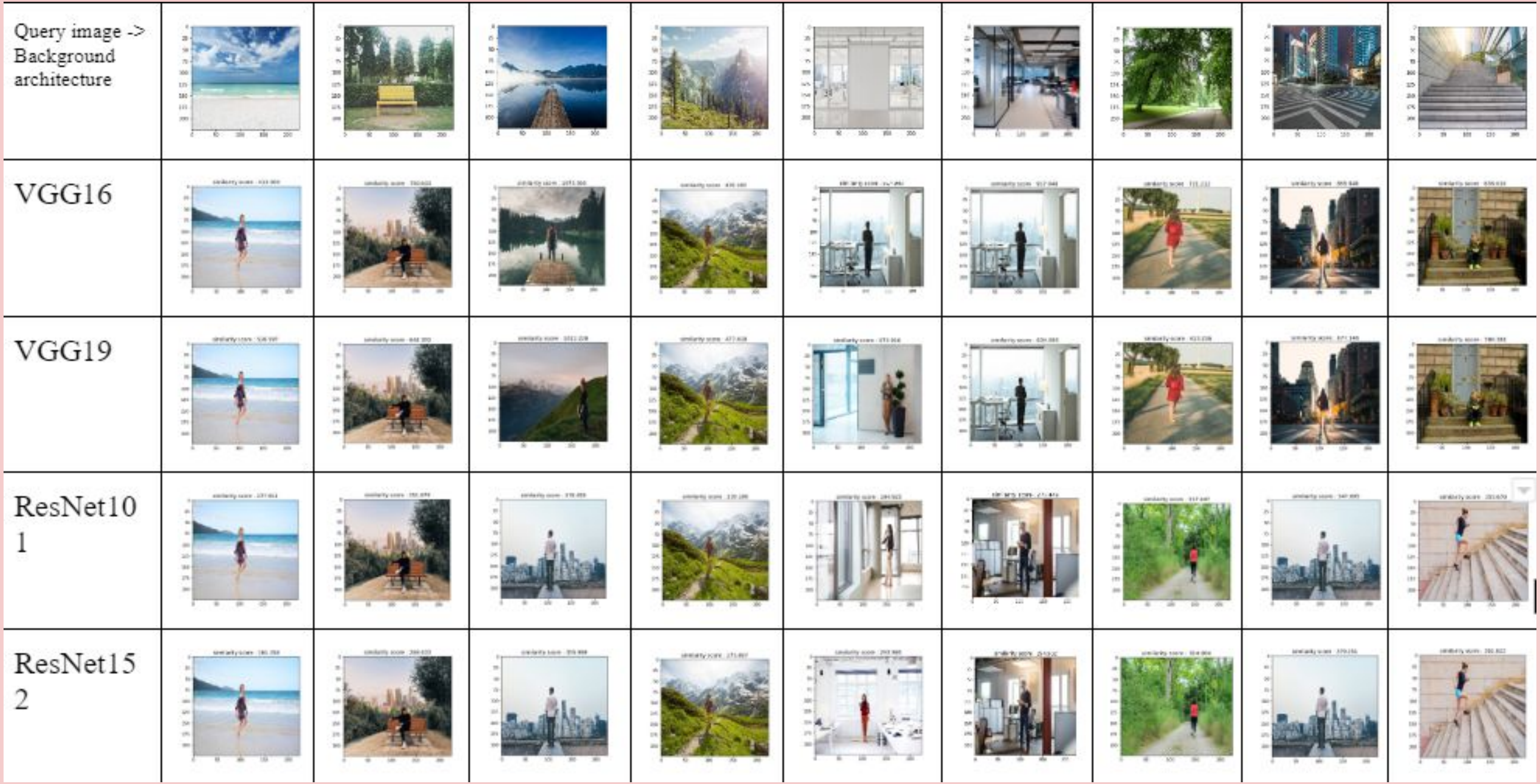
- Global features are extracted using CNN models, by stopping propagation at an arbitrary, but pre-specified layer.
- Extract the values from the specified layer and treat them as feature vector.
- We used several CNN models, out of which top 4 were observed as following:
    1. VGG16,
    2. ResNet101,
    3. VGG19,
    4. ResNet152.

- Obtained local and global feature vectors are then fused to get better representation of image.

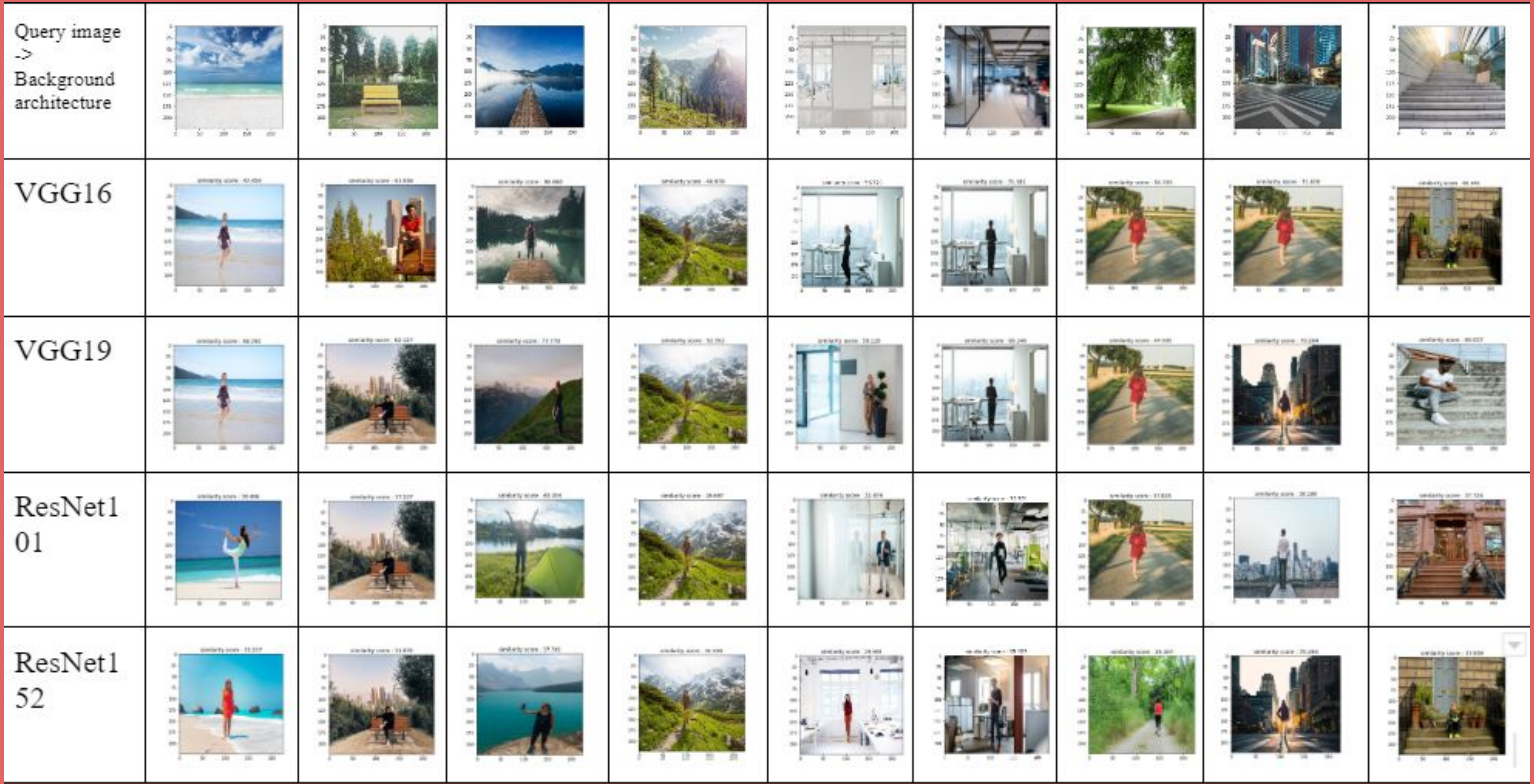
Results using Cosine similarity



Results using Chi square distance



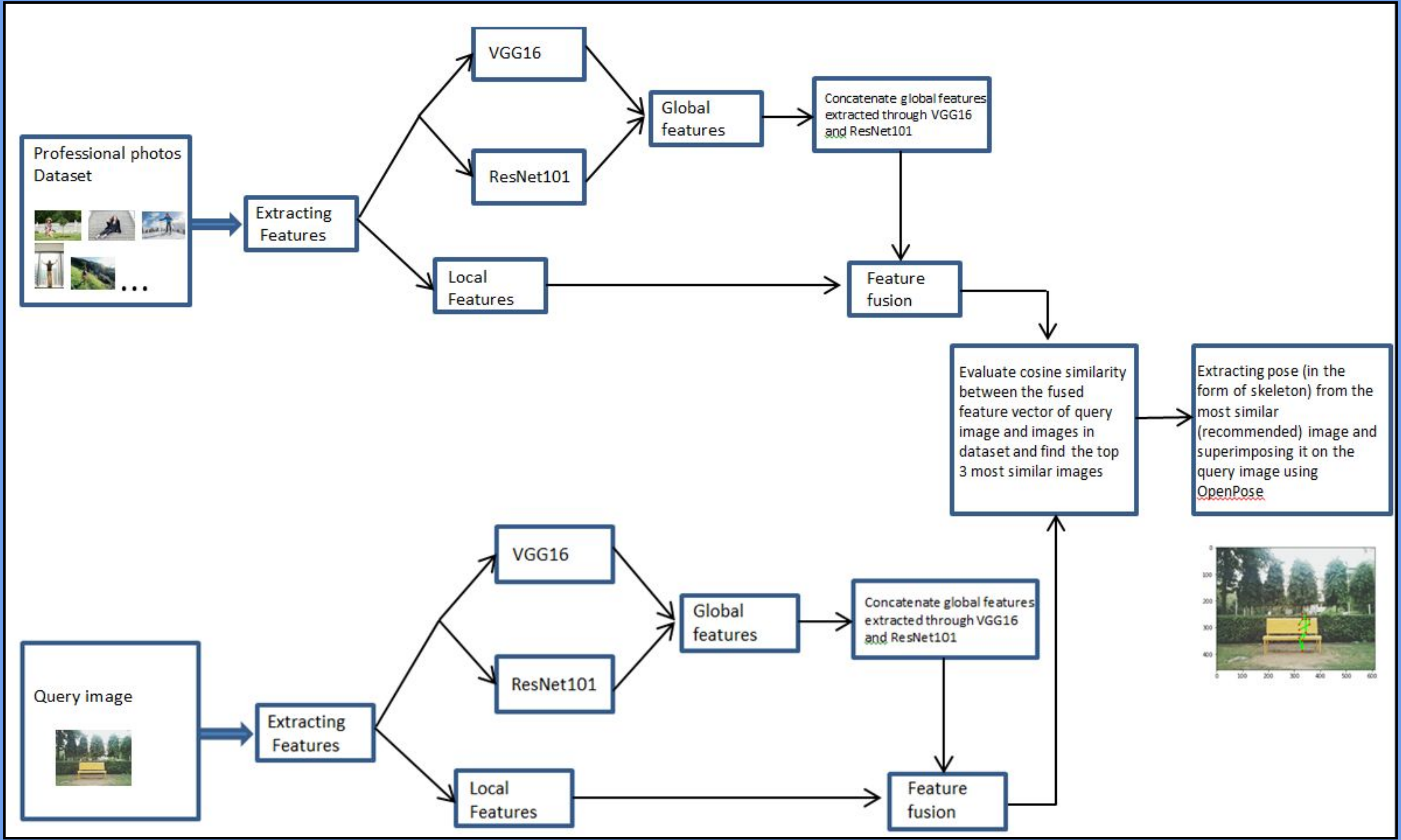
Results using Euclidean distance



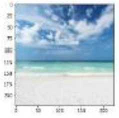
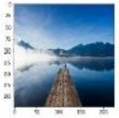
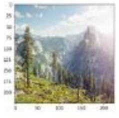
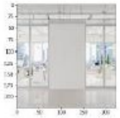
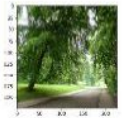
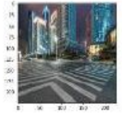
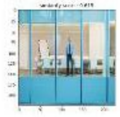
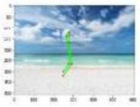
Proposed Method

- From the obtained results, it was observed that VGG16 and ResNet101 gave best results, and with similarity metric as Cosine similarity.
- Hence, we built hybrid model.
- It uses global features extracted using VGG16 (4096 dim) and ResNet101 (2048 dim) and concatenate them (6144 dim).
- Concatenated global features are fused with color histogram based local features (1440 dim).
- Final feature vector is of 7584 dimension.
- Cosine similarity is used to recommend best matching image to the query image form dataset.
- Pose from the recommended image is extracted and superimposed on query image using OpenPose.

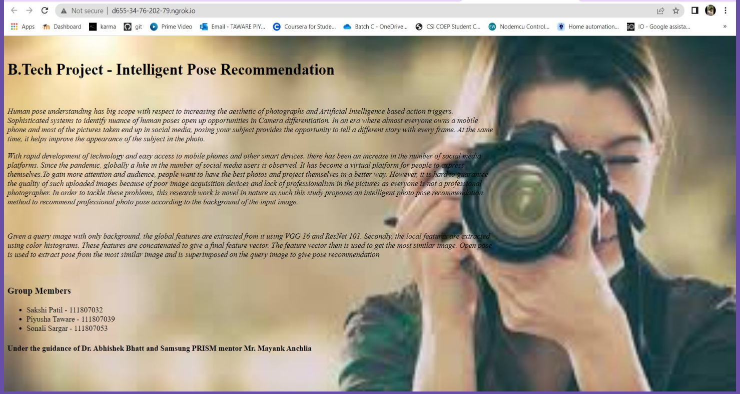
Framework of Proposed Method

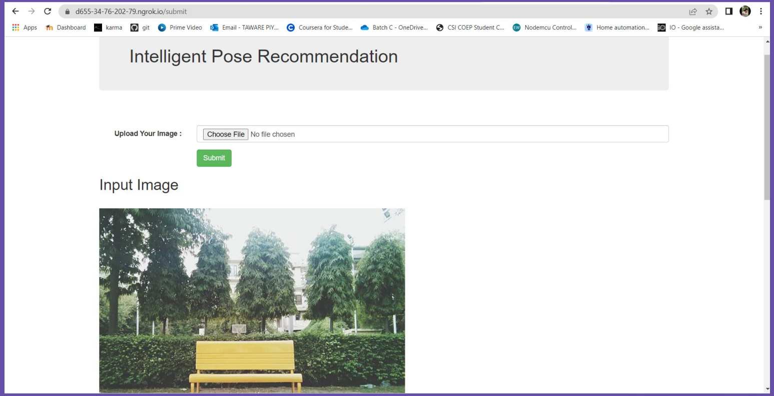


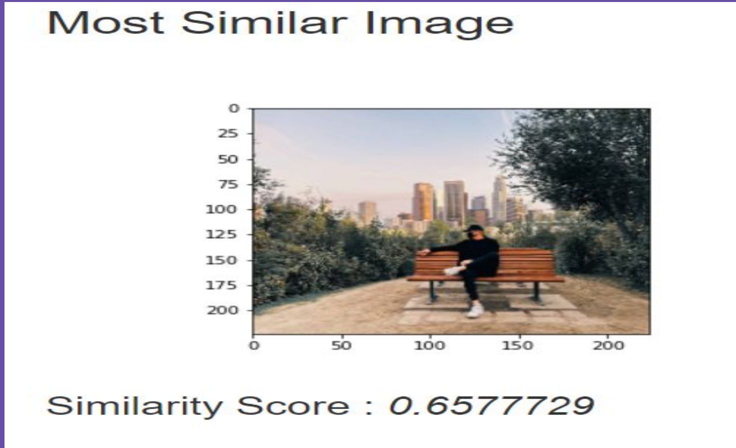
Results of proposed method

Query image ->									
Top recommendation by Hybrid Model (VGG16 and ResNet101)									
Pose recommendation based on 1st recommended image									

Web interface







Conclusion

- In this study, a novel approach for recommendation of poses for the input query image is proposed.
- A combination of global and local features is used for representing an image and recommending the most similar image.
- This approach utilised the 2 best performing CNN architectures and optimised the results on appropriate similarity metric.
- The further scope of this approach includes recommending poses for real time query images uploaded from users' cameras.
- This approach is for single person pose recommendation, it can be further extended to multiple person pose recommendation.
- Also, a module for pose correction can be introduced which can act as a feedback for users after a user tries to imitate pose as recommended by this algorithm.
- Proposed method is quite satisfactory in this new domain of work, and can be further extended to get wondrous results.

Timeline

Sept - Oct'21

Problem understanding and studying CNN architectures

November'21

Literature review of pose estimation models and Wrote a review paper summarizing these models

Dec'21 - Jan'22

Dataset curation, Preliminary implementation

Feb'22

Implement classification based approaches

March-April'22

Build recomm-endation system Accuracy assessment and publishing paper

REFERENCES

[1] RMPE: Regional Multi-person Pose Estimation – Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, Cewu Lu

[2] DeepCut: Joint Subset Partition and Labeling for Multi Person Pose Estimation - Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P.V., & Schiele, B. (CVPR 2016)

[3] OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields

Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, Yaser Sheikh

[4] Mask R-CNN-Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick

[5] DeepPose: Human Pose Estimation via Deep Neural Networks-Alexander Toshev, Christian Szegedy

[6]Efficient Object Localization Using Convolutional Networks-Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, Christopher Bregler

[7] Stacked Hourglass Networks for Human Pose Estimation-Alejandro Newell, Kaiyu Yang, Jia Deng

[8] Human pose estimation via Convolutional Part Heatmap Regression Adrian Bulat and Georgios Tzimiropoulos

REFERENCES

[9] PersonLab: Person Pose Estimation and Instance Segmentation with a Bottom-Up, Part-Based, Geometric Embedding Model - George Papandreou, Tyler Zhu, Liang-Chieh Chen, Spyros Gidaris, Jonathan Tompson, Kevin Murphy

[10] Adaptive recommendation for photo pose via deep learning "Tong Hao1 · QianWang1 · DanWu1 · JinSheng Sun1,2"

[11]https://www.creativeboom.com/resources/20-of-the-best-websites-to-download-free-stock-photography-in-2018/

[12]https://stocksnap.io/

[13]https://www.pyimagesearch.com/2019/05/20/transfer-learning-with-keras-and-deep-learning/

[14] Intelligent Portrait Composition Assistance "Farshid Farhat, Mohammad Mahdi Kamani, Sahil Mishra, James Z. Wang"

[15] Aesthetic Composition Representation For Portrait Photographing "Yanhao Zhang, Xiaoshuai Sun, Hongxun Yao, Lei Qin, Qingming Huang"

THANK YOU!

Results of the Review Paper

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	PCKh @0.5	mAP @0.5
Alpha Pose	88.4	86.5	78.6	70.4	74.4	73.0	65.8	-	76.7
Deepcut	94.1	90.2	83.4	77.3	82.6	75.7	68.6	82.4	-
Deeper Cut	96.8	95.2	89.3	84.4	88.4	83.4	78.0	88.5	-
CPM	97.8	95.0	88.7	84.0	88.4	82.8	79.4	88.5	-
IEF	95.7	91.7	81.7	72.4	82.8	73.2	66.4	81.3	-
Stacked Hourglass	98.2	96.3	91.2	87.1	90.1	87.4	83.6	**90.9**	-
Open Pose	91.2	87.6	77.7	66.8	75.4	68.9	61.7	-	75.6
HRNet-W32	**98.6**	**96.9**	**92.8**	**89.0**	**91.5**	**89.0**	**85.7**	-	**92.3**
Dark Pose	97.2	95.9	91.2	86.7	89.7	86.7	84.0	-	90.6

Comparison of PCKh@0.5(single person) and mAP@0.5(multi-person) on MPII test set.

Results of the Review Paper

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	PCK@0.2
Deep Cut	97.0	91.0	**83.8**	78.1	91.0	86.7	82.0	87.1
Deeper Cut	97.4	**92.7**	87.5	**84.4**	**91.5**	89.9	87.2	90.1
CPM	**97.8**	92.5	87.0	83.9	**91.5**	**90.8**	**89.9**	**90.5**
IEF	90.5	81.8	65.8	59.8	81.6	70.6	62.0	73.1

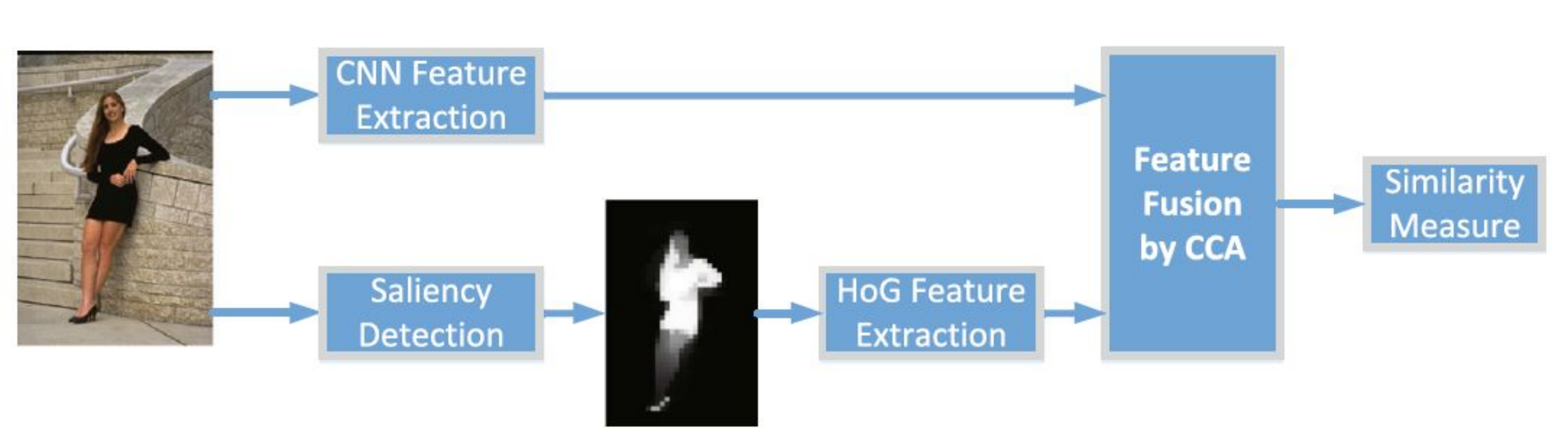
Comparison of PCK@0.2 on LSP test set.

Method	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}
Alpha Pose	73.3	89.2	79.1	69.0	78.6
Stacked Hourglass	71.3	90.1	78.0	67.3	77.3
Open Pose	61.8	84.9	67.5	57.1	68.2
HRNet-W48 + extra data	77.0	92.7	84.5	73.4	83.1
Dark Pose	**78.9**	**93.8**	**86.0**	**75.1**	**84.4**

Results on MSCOCO Keypoints Challenge(AP) dataset.

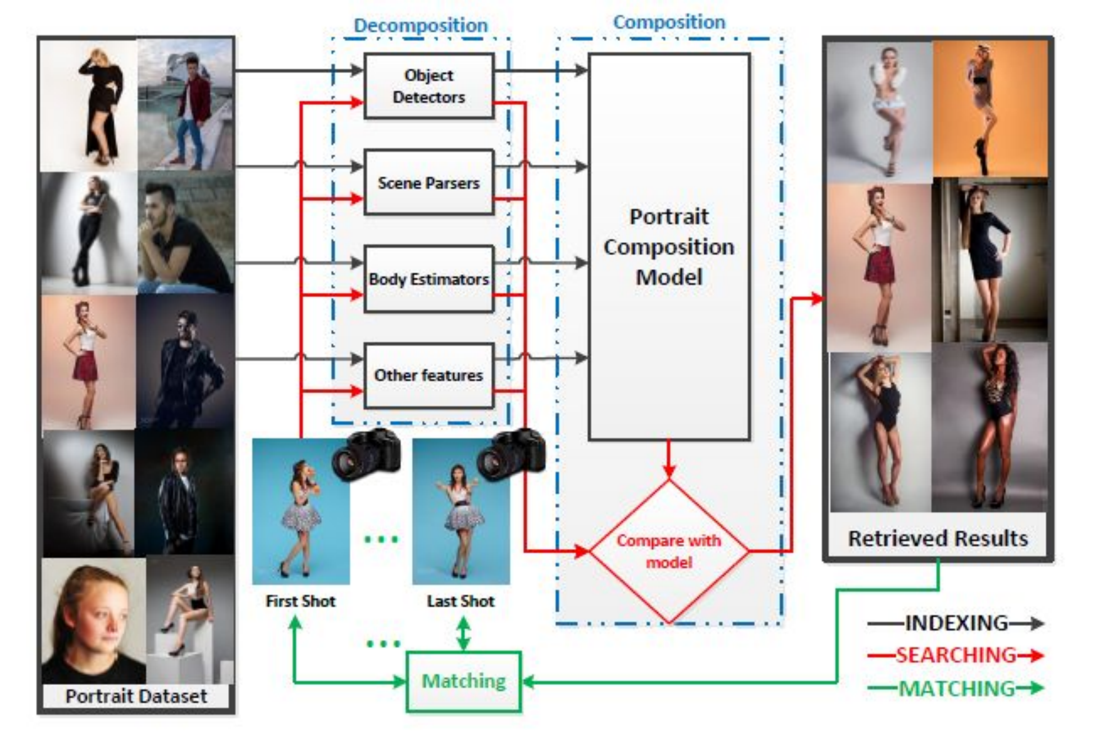
Literature Review

Adaptive recommendation for photo pose via deep learning [10]



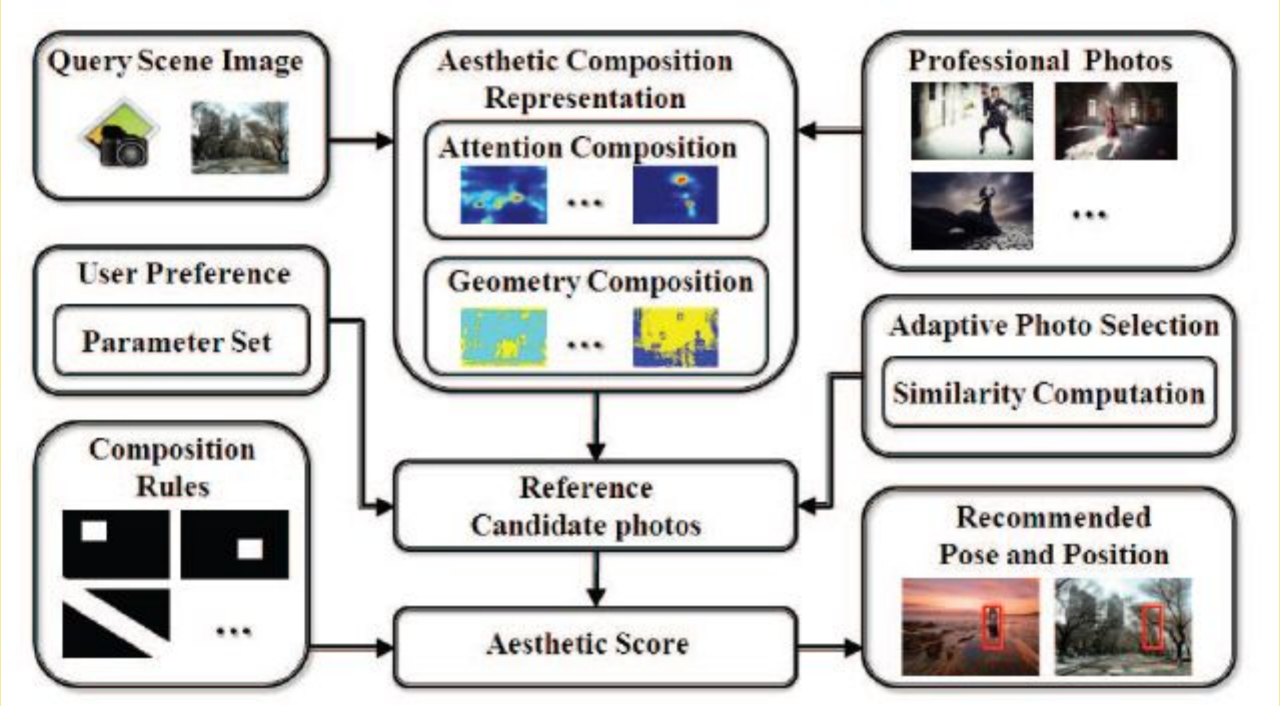
Literature Review

Intelligent Portrait Composition Assistance [14]



Literature Review

Aesthetic Composition Representation For Portrait Photographing [15]



Dataset

- There is no ready made dataset available for pose recommendation
- Previous works on this topic collect private dataset to evaluate their performance.
- We looked for professional photography websites for free downloading of photos
- We wrote a python script for crawling photos from a professional photo website, namely StockSnap.
- It contains photos from millions of creative photographers around the world.
- Also, we curated 400+ images spanning over 9 different classes based on image background.

Fig. : Dataset Generation

Implementation till now

Transfer learning Introduction

- Transfer Learning make use of knowledge gained while solving one problem and applying it to a different but related problem.
- This technique is used when we don’t have enough data for our problem statement, as in our case.
- There are two types of transfer learning in the context of deep learning:

1. Transfer learning via feature extraction
2. Transfer learning via fine-tuning

Extracting global features

- We are using the VGG16 architecture as a feature extractor, by chopping off fully connected layers.
- The last layer is Max Pooling layer, which has an output shape of 7*7*512.
- Flattening this volume into a feature vector, we obtain 7*7*512 = 25088 values for each image as our feature vector.
- Given N images, our dataset will now be represented as a list of N vectors each of 25088 dimension.

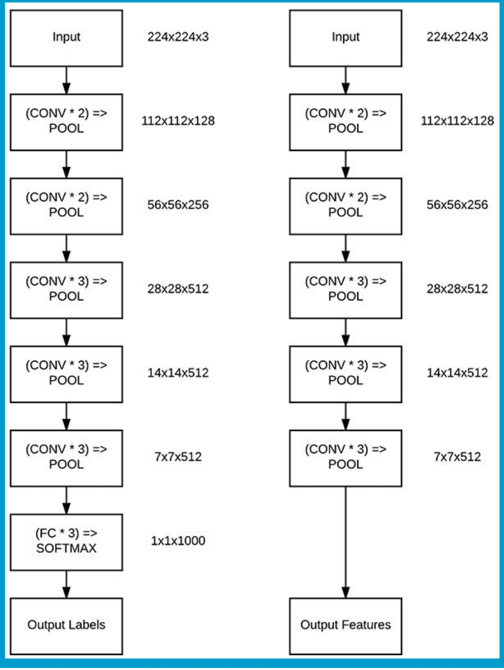


Figure 2: *Left*: The original VGG16 network architecture that outputs probabilities for each of the 1,000 ImageNet class labels. *Right*: Removing the FC layers from VGG16 and instead of returning the final POOL layer. This output will serve as our extracted features.

Data Preprocessing

- The website has photographs belonging to different categories like food, technology, nature etc
- Since our topic is human pose recommendation, we selectively crawled images which belonged to people, kids, women, men, family categories.
- We managed to collect around 414 images.(Can be further extended)
- We resize the image to 224*224 size for input to VGG16 model.
- The image is converted to array and mean RGB intensity is subtracted from it.