VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY-UNIVERSITY OF SCIENCE-INFORMATION TECHNOLOGY

LUU THI HONG NGOC- HO THIEN PHUOC

DIGITAL IMAGE PROCESSING- FACIAL ATTRIBUTE RECOGNITION

|| Guided by ||

Mr. LY QUOC NGOC

Report

CATALOGUE

- Motivation
- Problem Statement
- Related Works
- Result comments
- Personal information (Group ID, Student Id, Full Name)
- Responsibility
- Reference

CONTENT

PERFORMANCE TABLE

No	Method name + Framework (year)	Performance + Measure + Time + Complexity	Data	Advantages	Disadvantages
1	LNets (5 Conv. layers) and ANet (4 Conv. layers) 2015 1.LNet locating the entire face region in a coarse-to-fine manner is pre-trained with one thousand object categories of ImageNet and fine-tuned by image-level attribute tags 1.ANet is pre-trained by distinguishing massive face identities 2.ANet extracts a set of feature vectors 3.Averaging all attribute values given each FC	87% (Avg. of 40 attribute For a 300 * 300 image, LNets takes 35ms to localize face region while ANet takes 14ms to output extracted features on GPU	CelebA	LNet: significantly reduces data labeling, improves the accuracy of face localization ANet: reduce redundant computation in the feature extraction stage =>Save redundant computation, which enables evaluating image with arbitrary size in realtime. It allows taking images of arbitrary sizes	Small misalignment of face localization. Require expensive pre-training. Require part extraction steps.

				as input without normalization provides more accurate face localization, leading to good performance in the subsequent attribute prediction	
2	LMLE (large margin local embedding) 2019 1. Cluster for each class by k-means using the learned features from previous round of alternation. For the first round, we use hand-crafted features or prior features obtained from other pre-trained network. 2. Generate a quintuplet table using the cluster and class labels from a random subset (50%) of training data. 3. For CNN training, repeatedly sample mini-batches equally from each class and retrieve the corresponding quintuplets from the offline table. 4. Feed all quintuplets into five identical CNNs to compute the loss with cost-sensitivities. 5. Back-propagate the gradients to update the CNN parameters and feature embeddings. 6. Alternate between steps 1-2 and 3-5 every 5000 iterations until convergence (empirically within 4 rounds when we observe no improvements on validation set).	84% (Avg. of 40 attributes) 10ms per sample to extract features number of all clusters (⌊L/l⌋) with a complexity of O(L/l log(L/l)	CelebA	More resistance to data imbalance.Edge detection results outperform by a large margin. Accurately discover fine rare edges as well as the majority nonedges that make edge maps clean. Not suffer from the prediction bias with noisy edges and relatively low recall of fine edges respectively. Richer information and a stronger constraint than the conventional class-level image similarity. No information loss unlike under-sampling . No artificial noise unlike over-sampling.	Require expensive pre-training. Exponential growth of quintuplet number. Potential penalty inconsistency between sampled quintuplets. Enforcing Euclidean distance on a hypersphere manifold is not natural
3	Multi-task Restricted Boltzmann Machines with PCA and keypoint features 2016	87% (Avg. of 40 attributes)	CelebA	Achieved good results on vision problems such	NIL

	1.Regularizing the parameter space 2.Correlating relevant features jointly			as: person re-identification , multiple attribute recognition, and tracking	
4	Multi-task CNN features (3 Conv. layers and 2 FC layers) 2016 1.Training a single attribute network which classifies 40 attributes, sharing information throughout the network. 2.Learning the relationship among all 40 attributes, not just attribute pairs.	91% (Avg. of 40 attributes). The independent CNNs each take about an hour	CelebA	Require no expensive pre-training, alignment, or part extraction steps. Significantly decrease the number of parameters- over 4 times	NIL
5	Slim-CNN 2019 1. Multiple Branches 2. Small Kernels 3. Skip-Connections	91.24%	CelebA	Reduces the memory storage requirement. Trained on large-scale labeled data. Significantly reducing the number of parameters. Get superior performance while still keeping the computational complexity in check.	Less efficencent with limited labeled or unlabeled data
6	Deep multi-task learning (DMTL) 2016 1.The shared feature learning for all the attributes. 2.Category-specific feature	92.1% (Avg. of 40 attributes). 8ms on a Titan X GPU, and 35ms on an Intel Core I7 3.6 GHz CPU	CelebA	Individual attribute groups which are not well separable from each other in the original image space could become separable in the feature space learned by MTL, leading to improved	NIL

	learning for heterogeneous attribute categories.			multi-attribute estimation accuracy.Achieves promising attribute estimation accuracy while retaining low computational cost, making it of value in many face recognition applications.	

RELATED WORKS

There are large bodies of work on CNNs, Multi-Task Learning, and Attributes. We draw from all three areas to design the proposed method, MCNN-AUX. The relevant literature is reviewed in the following sections.

CNN

Deep CNNs have been widely used for feature extraction and have shown great improvement over hand-crafted features for many problems including object recognition, automatic caption generation, face detection, face recognition and verification, and activity recognition. CNNs have quickly gained popularity since the introduction of open-source software tools which allow for straight-forward construction, training, and testing of deep CNNs. Caffe, Torch, and TensorFlow are among the most popular packages for implementing CNNs. The first big success for deep CNNs in a large-scale problem was in the 2012 Imagenet Challenge with a network that outperformed the then existing methods [8]. Since then, a wide variety of CNN architectures have been proposed for many computer vision problems. CNNs have also dominated the field of face recognition and verification. One of the most notable works in this domain is that of Deep-Face, which utilized a large dataset and applied both a siamese deep CNN and a classification CNN in order to maximize the distance between impostors and minimize the distance between true matches. Motivated by the success on the LFW dataset, researchers focused more on CNNs for face recognition and the networks became deeper and more complex. In this work, we take advantage of the discriminative power of the CNN to learn semantic attribute classifiers as a mid-level representation for subsequent use in recognition and verification systems.

Multi-Task Learning

Multi-task learning (MTL) is a way of solving several problems at the same time utilizing shared information. MTL has found success in the domains of facial landmark localization, pose estimation, action recognition, face detection, and many more. Attributes and object classes are learned jointly to improve overall object classification performance. Wang, G. and Forsyth, D. use Multiple Instance Learning to detect and recognize objects in images by learning attribute-object pairs. Wang, Y. and Mori, G. uses an undirected graph to model the correlation amongst attributes in order to improve object recognition. In “Sharing features between objects and their attributes. CVPR”, attributes and objects share a low-dimensional representation allowing for regularization of the object classifier. In our work, all attributes share the lower layers in the CNN, so that information common to all the attributes can be learned. Applying MTL to attribute prediction is very natural given the strong relationships among the facial attributes

Attributes

Kumar et al. introduced the concept of attributes as image descriptors for face verification. They used a collection of 65 binary attributes to describe each face image. They later extended this work with an addition of 8 attributes and 4 Emily M. Hand and Rama Chellappa applied their method to the problem of image search in addition to face verification. Berg et al. created classfiers for each pair of people in a dataset and then used these classifiers to create features for a face verification classifier. Here, rather than manually identifying attributes, each person was described by their likeness to other people. This is a way of automatically creating a set of attributes without having to exhaustively hand-label attributes on a large dataset. Prior to this, there were decades of research on gender and age recognition from face images. CNNs have been used for attribute classification recently, demonstrating impressive results. Pose Aligned Networks for Deep Attributes (PANDA) achieved state-of-the-art performance by combining part-based models with deep learning to train pose-normalized CNNs for attribute classification. Focusing on age and gender, Levi, G. and Hassner, T. applied deep CNNs to the Adience dataset. Liu et al. used two deep CNNs - one for face localization and the other for attribute recognition - and achieved impressive results on the new CelebA and LFWA datasets, outperforming PANDA on many attributes. Unlike these methods, our MCNN-AUX requires no pre-training, alignment or part extraction. Past work has generally considered attributes to be independent, with “Pose aligned networks for deep attribute modeling. CVPR”, “ Attribute and simile classifiers for face verification. ICCV”, and “Deep learning face attributes in the wild.

ICCV" training a separate classifier for each attribute. Siddiquie, B., Feris, R.S. and Davis, L.S. use the correlation amongst attributes to improve image ranking and retrieval. They use independently trained attribute classifiers and then learn pairwise correlations based on the outputs of these classifiers. Our method goes above and beyond this by training a single attribute network which classifies 40 attributes, sharing information throughout the network, and by learning the relationship among all 40 attributes, not just attribute pairs. Abdulnabi, A.H., Wang, G., Lu, J. and Jia, K. used a multi-task network to learn attributes for animals and clothing, rather than faces. They utilize groupings as in "Decorrelating semantic visual attributes by resisting the urge to share. CVPR", but they impose constraints at the feature level according to the groups. We incorporate groupings directly into the network by sharing layers amongst attributes in a single grouping.

SOLUTION

Architecture

Conv1 consists of 75 7x7 convolution filters, and it is followed by a ReLU, 3x3 Max Pooling, and 5x5 Normalization. Conv2 has 200 5x5 filters and it is also followed by a ReLU, 3x3 Max Pooling, and 5x5 Normalization. Conv1 and Conv2 are shared for all attributes. After Conv2, groupings are used to separate the layers. There are nine groups in all: Gender, Nose, Mouth, Eyes, Face, AroundHead, FacialHair, Cheeks, and Fat. There are 6 Conv3s: one each for Gender, Nose, Mouth, Eyes, and Face, and one for the remaining groups - Conv3Rest. Each Conv3 has 300 3x3 filters and is followed by a ReLU, 5x5 Max Pooling and 5x5 Normalization. The Conv3s are followed by fully connected layers, FC1. There are 9 FC1s - one for each group. Each FC1 is fully connected to the corresponding previous layer, with Conv3Rest connected to the FC1s for AroundHead, FacialHair, Cheeks, and Fat. Every FC1 has 512 units and is followed by a ReLU and a 50% dropout to avoid overfitting. Each FC1 is fully connected to a corresponding FC2, also with 512 units. The FC2s are followed by a ReLU and a 50% dropout. Each FC2 is then fully connected to an output node for the attributes in that group. The attributes for each group are listed below:

– Gender: Male

– Nose: Big Nose, Pointy Nose

– Mouth: Big Lips, Smiling, Lipstick, Mouth Slightly Open

– Eyes: Arched Eyebrows, Bags Under Eyes, Bushy Eyebrows, Narrow Eyes, Eyeglasses

– Face: Attractive, Blurry, Oval Face, Pale Skin, Young, Heavy Makeup

– AroundHead: Black Hair, Blond Hair, Brown Hair, Gray Hair, Earrings, Necklace, Necktie, Balding, Receding Hairline, Bangs, Hat, Straight Hair, Wavy Hair

– FacialHair: 5 o'clock Shadow, Mustache, No Beard, Sideburns, Goatee

– Cheeks: High Cheekbones, Rosy Cheeks

– Fat: Chubby, Double Chin

The 9 groups were manually chosen according to attribute location. Some groupings were separated from others and some were absorbed into others through experimentation giving the above groupings. Male was kept separate from all other attributes as we found, through experimentation on the CelebA dataset, that gender was improved by sharing layers with other attributes, but it ultimately decreased performance of those attributes. We found the best compromise was to include male in the shared Conv1 and Conv2 layers and then to have separate Conv3, FC1, and FC2 layers. We use the Caffe software for our implementation, training, and testing of MCNN and MCNN-AUX [16]. We use a sigmoid cross-entropy loss applied to all attribute scores to facilitate training. As preprocessing steps, the training mean is subtracted from the images and they are cropped randomly with a size of 227x227. This helps the network to be robust to shifts in the input. If we were to use an independent CNN for each attribute following the architecture of one path in the MCNN - 3 convolutional layers and 3 fully connected layers - each CNN would have over 1.6 million parameters. So, for all 40 attributes, there would be over 64 million parameters. Using MCNN, we cut this down to less than 15 million parameters, over four times fewer.

MCNN-AUX

After training the MCNN, we add a fully connected layer, AUX, after the output of the trained MCNN. Starting with the weights from the trained MCNN, we learn the weights for the AUX portion of the network, keeping the weights from the MCNN constant. The AUX layer allows for interactions amongst attributes at the score level. The MCNN-AUX network learns the relationship amongst attribute scores in order to improve overall classification accuracy for each attribute. Figure 2 shows the connection between MCNN and AUX. The AUX layer only adds 1600 parameters to the 1.6 million from MCNN.

RESULT COMMENTS

We present results for our independent CNNs, MCNN, and MCNN-AUX. For comparison, we also provide the state-of-the-art by Liu et al., and a baseline of always choosing the most common label for each attribute. We can see from Table 1 that our independent CNNs outperform Liu et al. on most attributes for CelebA. The independent CNNs improve on Liu et al. by 15% for necklace, 12% for blurry, 9% for straight hair, and 8% for big nose. MCNN makes even further improvements, and finally MCNN-AUX gives the highest accuracy for most attributes. We see that the largest jump in performance is from the method of Liu et al. to the independent CNNs, with smaller improvements being made with MCNN and MCNN-AUX. From this, we see that the value in MCNN and MCNN-AUX is in the increased training speed and the decreased parameters, which reduces the chances of overfitting. We do not expect to see an increase in performance with MCNN-AUX for every attribute, as many attributes do not have strong relationships with the others. Determining which relationships to use can be done in the validation portion. We did not remove any relationships in our testing. Unlike Liu et al., all three of our methods outperform the baseline for every attribute in CelebA. Figure 4 shows a heatmap of the weights for the AUX layer of MCNN-AUX on the CelebA dataset. From Figure 4 we can see that each attribute contributes the most to its final classifier score. Some interesting relationships can be seen in the heatmap. We see that bald is strongly related to receding hairline and has an inverse relationship with straight hair and wavy hair and that no beard has an inverse relationship with 5 o'clock shadow, mustache, and sideburns. The strongest relationships are summarized in Table 2. Most of the relationships listed in Table 2 make intuitive sense. Someone with heavy makeup is likely to be wearing lipstick; if someone is chubby, they likely have a double chin; and if someone has gray hair, it is unlikely that they are young.

CONCLUSION

- Through this course, we gained more knowledge about applying the lessons in theory class. We know one more applications in graphics.
- There are many other projects around, so the time spent on this project is not enough to develop further. Within the scope of the project, we also tried the possible methods and referred to many other articles/documents.

- To complete this lab, thanks to the teachers with the enthusiastic and dedicated teaching assistants being so kind with us in this subject, we wish you health and joy in your work-teaching at HCMUS.