

000
001
002
003
004
005
006
007
008
009
010
011

Abstract

We explore the use of pre-trained convolutional neural network features for social group recognition. A social group recognition framework is proposed. It takes in both individual and global features of group images, and shows promising results for group recognition task which significantly outperform previous results. We verify the generalization ability and semantic information of pre-trained CNN features. We show the necessity of fine-tuning in social group recognition, and this suggests a potential usage of pre-trained CNN features with adaptation for novel computer vision tasks.

1. Introduction

In the past few years, there have been impressive progress in understanding semantic meaning of images, such as object recognition, scene recognition, and object detection. The power of Convolutional Neural Networks (CNNs) has especially noticeable. However, studies in analysis of a group of people in images are still deficient. Current search algorithms fail to capture information of personal styles or social characteristics of groups of people, but retrieve images with similar global appearance [8]. The analysis of groups of people is difficult in that the group categories are semantically ambiguous, and have high intra-class variance.

Recognition of groups of people from a social perspective provides many potential applications. With more accurate group searching results, more accurate recommendations can be made in social networks, and more relevant advertisement for particular groups of people benefits both consumers and sellers.

[12] and [8] studied this problem of group recognition. They created an urban tribe dataset consisting of 11 categories, with about 100 labeled per class. They proposed a group recognition pipeline. Rather than classifying isolated individuals in the group images, they focused on group features and models.

CNN architecture has been proved to achieve outstand-

Urban Tribe

Anonymous WACV submission

Paper ID ****

ing results in various computer vision tasks, and it's argued that deep architectures in CNN can capture visual features of different semantic level in the hidden units ([?]). Recently, features learnt for large scale recognition tasks with large amounts of training data have been used for new tasks, and the features outperform many conventional features in many tasks([?], [2]). This verifies that generic visual features are obtained from pre-trained CNN model.

this is what to modify later with results The most frequently used pre-trained model was proposed in [?] trained with ImageNet¹. And the new tasks for which CNN features are used range from object recognition, scene recognition, to subcategory recognition. However, the objects or categories of these tasks are mainly covered in the 1000 ImageNet categories, and it is predictable that some features pre-trained on ImageNet are useful for the new tasks. Meanwhile, the social groups of individuals require semantic features of individual's style, which have little overlap with the 1000 ImageNet categories.

In this paper, we investigate the generalization ability of pre-trained CNN features to social group recognition. We propose a CNN feature based architecture for social group recognition. Our model takes in both individual feature and global scene feature finetuned from CNN pre-trained weights. Our result shows a boost of performance from the previous group classification method provided by [8]. We show that both individual information and global scene information contribute to a social group's characteristics, and that different feature extraction schemes for individual and global information is necessary. We also show the role of adapting generic pre-trained CNN features to social group images with finetune.

Further, we visualize the CNN features in different layers, and show a better semantic clustering with ascending layer, validating the feature composition of our framework. Finally, we analyze the properties of the social groups and the urban tribe dataset, showing that the representation our model achieves is in a high semantic level.

¹<http://image-net.org/challenges/LSVRC/2012/browse-synsets>

108 **2. Related Work**

109 Convolutional Neural Networks(CNNs) with back-
 110 propogation were introduced around 1990's by [10]. Since
 111 then, CNNs showed successful results on various computer
 112 vision tasks, such as hand-written digit classification ([11]),
 113 classification task on ImageNet dataset ([?]).

114 Recently, many researches show the ability of generaliza-
 115 tion of pre-trained CNN features on large dataset such
 116 as ImageNet. [?] shows excellent generalization of the
 117 pre-trained CNN features. They kept all the layers of
 118 ImageNet-trained model fixed except last softmax classi-
 119 fier, and achieved bets results on Caltech-101 and Caltech-
 120 256. [2] used different layers of pre-trained CNN network
 121 as features and trained simple classifiers such as SVM and
 122 Logistic Regression, and outperformed the state-of the-art
 123 on several vision challenges such as scene recognition and
 124 domain adaptation. GoogLeNet ² used improved CNN ar-
 125 chitecture pre-trained on ImageNet dataset, and won the the
 126 detection task of Large Scale Visual Recognition Challenge
 127 2014 (ILSVRC2014).

128 [8] created an urban tribe dataset consisting of 11 classes.
 129 The classes are defined from social group labels provided
 130 by Wikipedia. They selected the eight most popular cate-
 131 gories from their list of subculture, and added three other
 132 classes corresponding to typical social venues in addition.
 133 For each classes, images of groups of people are searched
 134 with different search engines, and a broad range of scenar-
 135 os for each class are collected. [12] and [8] also provide a
 136 group description and several classification methods. Group
 137 description consists of person descriptors and global group
 138 descriptors: Six part of person is detected, and a set of pre-
 139 defined descriptors are computed for each part; Global
 140 descriptors use a both low level and high level descriptors to
 141 describe the context and group properties of the image. Two
 142 options of classification methods are provided: bag of parts-
 143 based classification and SVM-based classification.

144 Categorizing the social groups of individuals belongs to
 145 fine-grained classification task which recently draws more
 146 interest of the computer vision community. It aims at giving
 147 the fine-grained categories in a certain class. Fine-grained
 148 classification is more difficult than conventional classifica-
 149 tion tasks, because the categories are semantically as well as
 150 visually similar, and are even challenging for humans. The
 151 Fine-grained Challenge 2013 (FGComp) provided the data
 152 in several categories including aircraft, birds, cars, dogs,
 153 and shoes. [5] achieved the best result using classifier based
 154 on fisher vectors. However, CNN based methods using [6]
 155 or [2] gave not so good results, especially when the bound-
 156 ing box of test data is missing.

157 There is some research in analyzing social groups of peo-
 158 ple. [4] showed the visual structure of a group helps under-

159 ²<http://www.image-net.org/challenges/LSVRC/2014/results>

160 standing events. [13] showed social relationships modeling
 161 helps people recognition. [1] used both local and global
 162 factors for group level expression analysis.

163 **3. Methods**

164 This section describes the urban tribe dataset and elabo-
 165 rates on the model architecture.

166 **3.1. Urban tribes dataset**

167 Urban tribes are groups of people who have similar vi-
 168 sual appearances, personal style and ideals. The urban
 169 tribes dataset consists of 11 different categories: biker,
 170 country, goth, heavy-metal, hip-hop, hipster, raver, surfer,
 171 club, formal, casual/pub, with an average of 105 images
 172 from each category.

173 Different from conventional visual classification prob-
 174 lem, urban tribe categories are more ambiguous and sub-
 175 jective. Also, each class contains a broad range of scenar-
 176 os. The high intra-class variation of the urban tribe dataset
 177 makes the classification task challenging. The urban tribe
 178 dataset also has some interesting properties. The number
 179 of people in each urban tribe image varies. Members in one
 180 tribe often have similar visual styles, including their clothes,
 181 accessories, and even demeanor. For example, surfers pos-
 182 sibly carry surfboards, and the goth often have dark attire,
 183 makeup and hair. The environment they are in also con-
 184 tributes to each tribe characteristics: pictures of countries
 185 are more likely to be taken outdoor with grassland, while
 186 pictures of clubbers are often photographed in clubs with
 187 dim lightings.

188 **3.2. Classification hierarchy**

189 To utilize the properties of urban tribes fully, our feature
 190 vector consists of both elements: individual features and
 191 environment features. For each feature type, we use simi-
 192 lar extraction strategy. Individual features and environment
 193 features are hierarchically combined to form the final deci-
 194 sion function. The network hierarchy is shown in Figure 1.

195 For each group image, we represent the group G as the
 196 combination of a set of people and the environment. To give
 197 the prediction of class C , the individual feature vectors and
 198 environment feature vectors are extracted separately.

199 For the individual feature vectors, first, individual per-
 200 son candidates are detected with a poselet based person
 201 detection algorithm. The person candidate images $H =$
 202 $\{H_1, H_2, \dots, H_p\}$ are used as a whole instead of a set of
 203 body part bounding boxes. Each person candidate is re-
 204 sized to 256×256 , and ten 227×227 patches $\{h_{ij}\}, i \in$
 205 $\{1, 2, \dots, p\}, j = 1, 2, \dots, 10$ are extracted using the same
 206 method as fine-tune image set generation.

207 Each Individual image patch h_{ij} then passes through
 208 the Convolutional Neural Network for person images

216 CNN_{Person}, generating activations from the 6th and 7th
 217 hidden layer. The activations from 6th and 7th layer are
 218 both in 4096 dimensions. They are concatenated to form
 219 a 8192-dimensional vector f_{ij} , where $i \in \{1, 2, \dots, p\}, j \in$
 220 $\{1, 2, \dots, 10\}$.

221 The feature vectors are then fed into a multi-class
 222 SVM_{Person}. We use LIBLINEAR[3] to train the SVM
 223 on individual patches, and to estimate probabilities for
 224 each category given individual patch h_{ij} : $\Pr_{ij}(C|h_{ij}), C \in$
 225 $\{1, 2, \dots, c\}$, where c is the number of classes in urban tribe
 226 dataset. The individual patches h_{ij} in one group image are
 227 usually highly correlated. Therefore, in order to obtain a re-
 228 liable probability estimate from the noisy yet correlated set
 229 of probabilities \Pr_{ij} , a simple but effective average pooling
 230 is performed to \Pr_{ij} :

$$\Pr_{People}(C|H_1, \dots, H_p) = \frac{1}{10p} \sum_{i,j} \Pr_{i,j}(C|h_{ij}) \quad (1)$$

235 $\Pr_{People}(C|H_1, \dots, H_p)$ is the probability estimate of class
 236 C given the set of people candidate images H .

237 On the other hand, the entire environment in the scene
 238 image, denoted by S , is directly utilized for probability es-
 239 timation. The procedure to generate probability estimate of
 240 class C given the environment as a whole $\Pr_{Scene}(C|S)$ is
 241 similar with that for $\Pr_{People}(C|H)$. The difference is that
 242 the input of Convolutional Network is 227×227 patches
 243 extracted from the entire scene image, and the fine-tuned
 244 Convolutional network: CNN_{Scene} and SVM: SVM_{Scene}
 245 are trained with the training set of entire scene images. Sev-
 246 eral different strategies to extract patches from scene images
 247 and corresponding Convolutional Neural Network architec-
 248 tures are explained in Section 3.3.

249 Therefore, the probability estimate of a class C given
 250 observation of scene S is:

$$\Pr_{Scene}(C|S) = \frac{1}{K} \sum_{k=1}^K \Pr_k(C|s_k) \quad (2)$$

255 where K is the number of scene patches extracted from one
 256 group image, and this number varies with different patch
 257 extraction strategies. s_k is the k th scene patches. $\Pr_k(C|s_k)$
 258 is the probability for class C given k th scene patch. Average
 259 pooling is still used here, because the assumption of high
 260 correlation in patches holds.

261 Now we have the estimates of two kinds of conditional
 262 probability $\Pr_{People}(C|H)$ and $\Pr_{Scene}(C|S)$. We make a
 263 strong assumption that the two types of features are inde-
 264 pendent, and that the prior probability distribution of the
 265 urban tribes $\Pr(C)$ is a uniform distribution. The classifica-
 266 tion problem can be expressed as maximizing the objective
 267 function:

$$L = \arg \max_{i=1, \dots, c} \Pr(C = i|G) \quad (3)$$

where

$$\begin{aligned} \Pr(C = i|G) &= \Pr(C = i|H, S) \\ &= \frac{\Pr_{People}(C = i|H_1, \dots, H_p) \cdot \Pr_{Scene}(C = i|S)}{\Pr(C = i)} \\ &\propto \Pr_{People}(C = i|H_1, \dots, H_p) \cdot \Pr_{Scene}(C = i|S) \end{aligned} \quad (4)$$

and L is the predicted label for the group image.

3.3. Convolutional network feature extraction

It is shown in many experiments that a set of weights of convolutional network trained from ImageNet can generate a set of generic visual features.

Following [6]'s work, we use the network framework called Caffe. The network architecture is described in [?], which won the ImageNet Large Scale Visual Recognition Challenge 2012. We take the activations from the 6th and 7th hidden layer of the convolutional neural network, which are two fully connected layers before the class prediction layer. We also take the activations from 6th or 7th layer alone as comparison. We choose these two layers, because as the layers ascend, the features extracted show increasing invariance and semantic meaning.

We use the pre-trained set of weights of the network released by Caffe as the initial parameters of our network. The pre-trained model was trained on Imagenet ILSVRC-2012, and all images are first resized to 256×256 before they can be used as inputs to the network.

3.3.1 Pre-processing of the dataset

The urban tribe dataset is a relatively small dataset, and both people candidate crops and scene images are of various resolution. Our convolutional neural networks requires constant input image size of 227×227 , so pre-processing of the dataset is necessary.

There are several strategies to make one image compatible with the CNN:

1. As in [?], resize the image to a fixed resolution of 256×256 , and crop five 227×227 patches (from four corner and the center) and their horizontal reflections to generate ten patches from one single image. This way, the aspect ratio of the original images are lost, but for each crop, the portion it takes from the original image is fixed, so that the amount of information all the crops have is relatively stable.
2. Keep the aspect ratio of the original image, resize the shorter side to 256, and then crop five 227×227 patches and their horizontal reflections as mentioned. This method avoids distortion of the image and objects in it, but the crops will possibly lose much information

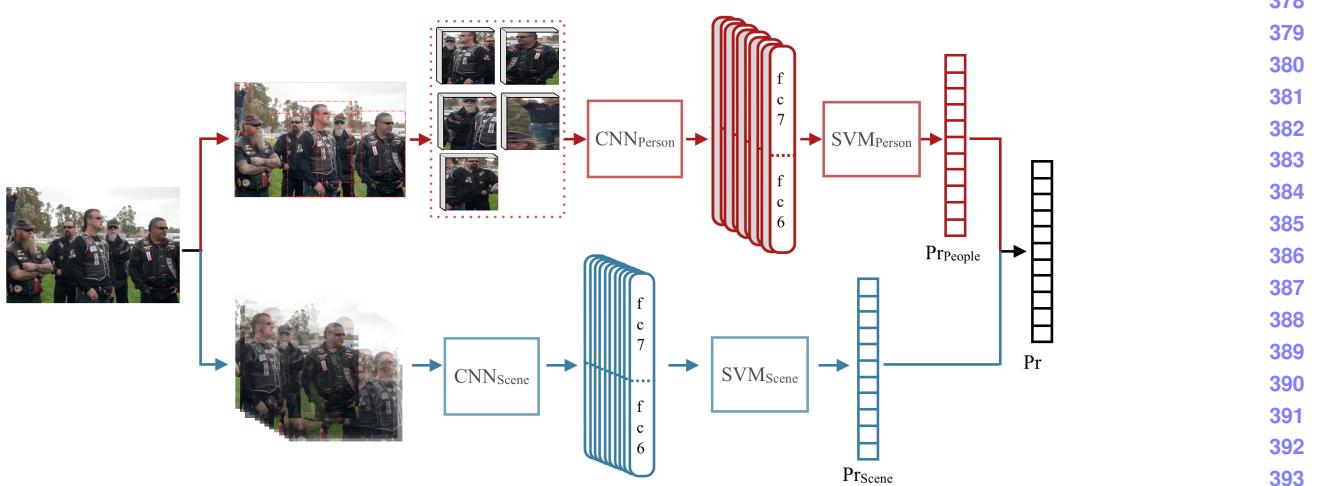


Figure 1: Architecture of classification algorithm using Nets_{SDense} . The upper half estimates the probability given people candidate images, and the lower half estimates the probability given the entire scene. Dense crop CNNScene and distorted crop CNN_{Person} are used.

when the aspect ratio of the original image is far away from 1.

- 347 when the aspect ratio of the original image is far away
348 from 1.
- 349
- 350 3. Keep the aspect ratio of the original image, resize the
351 shorter side to 256, and then densely crop multiple
352 227×227 patches and their horizontal reflections. This
353 way, the information of original image is kept by dense
354 cropping process, and the distortion is avoided. The
355 number of crops attained with this method is larger
356 than the previous two methods.

3.3.2 Network Finetune

Although the pre-trained network from [6] can already generalize well to many datasets, the urban tribe dataset has its unique property. It emphasizes certain visual features such as clothes, while pays less attention to other visual features. Also, it emphasizes style of the object rather than distinct category. To rearrange the importance of different features and adjust the features to adapt to urban tribe dataset, the network can be finetuned.

The dataset used for finetune is the same set used for future training. The input patches of the Convolutional network is of size 227×227 . The initial Convolutional network has 1000 outputs in final layer, corresponding to 1000 class-wise probability predictions. In our finetune process, the last layer is replaced by 11 probability prediction outputs, and the initial weights of the last layer connection are initialized to have zero mean gaussian distribution. Back propagation is used, and the learning rate is set to be small so that the fine-tune process adapts the extracted features

to urban tribe dataset while preserving the initial property in general: the initial learning rate used for pretraining is 0.01; We set the initial finetune learning rate of the parameters except for the last layer as 0.001, and keep the initial learning rate for last layer 0.01, because the last layer is not pre-trained.

3.3.3 Network modification

The side effect of dense cropping technique in Section 3.3.1 is that there is a boost in the number of input crops, and the speed of feature extraction is largely slowed down.

We apply a trick to speed up the process. In a convolutional neural network, a fully connected layer can be viewed as a convolutional layer with 1×1 sized kernels. Therefore, in our CNN architecture, we can substitute convolutional layers for fully connected layers of 6th and 7th layer. For example, the input of 7th layer is 4096-dimensional feature, and the output is 4096-dimensional feature. The layer has 4096×4096 connections. We can view the connections as 4096×4096 convolutional filters with size 1×1 . For one input image of size 227×227 , the output of the 7th layer is 4096 feature maps of size 1×1 , while for one input image of arbitrary size (larger than 227×227), the output of the 7th convolutional layer is 4096 feature maps of larger size, each 4096-dimensional element corresponds to one 227×227 cropping of the input image. Now the modified network can take images with arbitrary size (larger than 227×227) as input, and extract patch features much more efficiently.

432 **3.3.4 Choices of network combination**

433 Scene images and individual images have different properties,
 434 and need different strategies of pre-processing and separate fine-tuning.
 435 For scene network $\text{CNN}_{\text{Scene}}$ and scene images, due to the small size of the dataset, we use the dense
 436 cropping technique to increase the dataset. Only the center crops of each original group image and their mirrors are
 437 used for finetune. For person network $\text{CNN}_{\text{Person}}$ and corresponding input, we use the first cropping technique, because
 438 the subimages have normally long height and short width, and the second and third strategies using squared
 439 crops of a person image will lose much information, no matter which location we choose to crop them; whereas the first
 440 method ensure each crop keeps the essential features for classification. Finetuning of $\text{CNN}_{\text{Person}}$ uses center crops
 441 and their mirrors of person candidate images.

442 The combination of dense crop $\text{CNN}_{\text{Scene}}$ and distorted
 443 crop $\text{CNN}_{\text{Person}}$ are denoted as $\text{Nets}_{\text{SDense}}$.

444 We also construct other combination of networks for
 445 comparison:

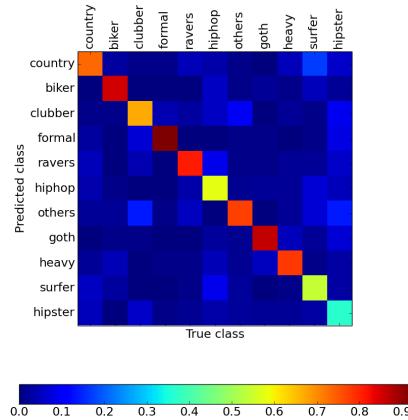
- 446 1. $\text{Nets}_{\text{NoTune}}$: Directly use the pre-trained network by
 447 [6] for both scenes and persons, and use the first cropping
 448 technique (distorted crops) as input patches for both networks. This choice of cropping strategy is in
 449 consistent with the way the network is pre-trained.
- 450 2. $\text{Nets}_{\text{SSparse}}$: Use the second cropping strategy for
 451 scene features, and the first cropping strategy for
 452 person features. Finetune procedure is the same as
 453 $\text{Nets}_{\text{SDense}}$.
- 454 3. $\text{Nets}_{\text{SDistort}}$: Use the first cropping strategy for both
 455 scene features and person features. Training set for
 456 finetune procedure is the center crops and their mirrors
 457 of scene/person images.

468 **4. Experiments and Results**

471 In this section, the performance of the proposed classification
 472 scheme is evaluated and analyzed. In the experiments, six rounds of 5-fold cross validation are performed,
 473 therefore we have 30 training experiments in total. Dataset
 474 is partitioned into 5 equal sized subsets, containing one-fifth
 475 of the data points from each category. One of the subsets is
 476 used as test set, and the remained 4 subsets are used as training
 477 data. In the finetune procedure, there are 6000 training
 478 iterations, and learning rate is decreased by ten times after
 479 each 1000 iterations.

480 **4.1. Urban tribe classification performance**

481 Table 1 shows the comparison of performance using different
 482 approaches. The 30 segmentations of datasets are used for all the approaches tested in this section, and 30 test



498 Figure 2: Confusion matrix for classification results with
 499 $\text{Nets}_{\text{SDense}}$, using people and scene features.

500 results are averaged for each approach. The standard error
 501 is shown with the accuracy in Table 1. We also compare our
 502 result with the result achieved by [8] using their best model.
 503 The advantage of CNN pre-trained features is obvious.

504 The confusion matrix is shown in Figure 2 for
 505 $\text{Nets}_{\text{SDense}}$, and all the 30 training experiments are aver-
 506 aged. We can observe there is a obvious difference of dif-
 507 ficulty of different categories. Class *formal* has accuracy as
 508 high as about 90%, while class *hipster* is the most difficult
 509 class, having less than 60% accuracy.

510 Comparing the result of using different features in the
 511 same approach shows the necessity of every step of our ar-
 512 chitecture. In results using $\text{Nets}_{\text{SDense}}$ with concatenated
 513 features, average accuracy for each individual person can-
 514 didate is low as 47.10%. Average pooling of individual
 515 candidate probability estimates produces a large accuracy
 516 increase of about 20%. Accuracy using the entire scene
 517 only results in 67.26% accuracy. Combining probabilities
 518 $\text{Pr}_{\text{People}}(C|H_1, \dots, H_p)$ and $\text{Pr}_{\text{Scene}}(C|S)$ achieves accu-
 519 racy as high as 71.22%, which verifies the complementary
 520 role of people candidate feature and environment feature in
 521 a group image.

522 We also compare vertically the accuracy of different ap-
 523 proaches, to show the role of network feature concatena-
 524 tion. Using only 7th layer or 6th layer activation from
 525 the networks $\text{Nets}_{\text{SDense}}$ produces decent results, showing
 526 both layers' activations can generate high semantic fea-
 527 tures. Concatenating both layers' activations increases the
 528 accuracy by 0.5%, indicating the slight information loss of the
 529 7th fully connected layer.

530 To see the role of finetuning, we can compare the result
 531 of $\text{Nets}_{\text{NoTune}}$ and $\text{Nets}_{\text{SDistort}}$. These two approaches

540 Table 1: Performance of different approaches using different information.
541

542 Accuracy (%)	543 Individual candidate	544 People	545 Entire scene	546 People+Scene
543 with concatenated features	544 39.99 ± 0.30	545 64.09 ± 0.63	546 62.77 ± 0.52	547 69.28 ± 0.50
544 Nets _{SDense} with fc7 features	545 47.07 ± 0.30	546 67.09 ± 0.52	547 65.05 ± 0.42	548 70.74 ± 0.47
545 Nets _{SDense} with fc6 features	546 45.84 ± 0.34	547 66.20 ± 0.46	548 67.08 ± 0.51	549 70.43 ± 0.46
546 Nets _{SDense} with concatenated features	547 47.10 ± 0.34	548 67.29 ± 0.50	549 67.26 ± 0.50	550 71.22 ± 0.46
547 Nets _{SSparse} with concatenated features	548 47.10 ± 0.34	549 67.29 ± 0.50	550 66.81 ± 0.42	551 71.23 ± 0.49
548 Nets _{SDistort} with concatenated features	549 47.10 ± 0.34	550 67.29 ± 0.50	551 65.35 ± 0.37	552 71.15 ± 0.50
549 SVM ₈ [8]	550 -	551 -	552 -	553 46(std: 2)

551 both use resizing that causes distortion, and they only vary
552 in finetune procedure. There is a large performance im-
553 provement with finetune, both for person features and scene
554 features. This shows the benefit of adapting existing generic
555 Convolutional Network to specific dataset.

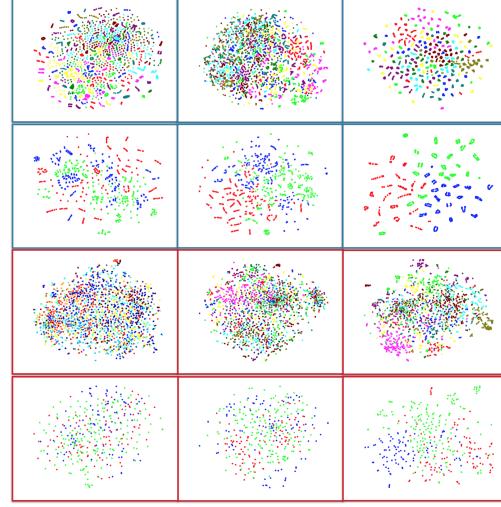
556 Nets_{SDistort}, Nets_{SSparse}, and Nets_{SDense} use different
557 patch extraction strategies. Note that we use the same dis-
558 torted patch extraction method for person images, as men-
559 tioned in Section 3.3.4, while we use three different meth-
560 ods for scene images. The results for scene images show
561 the advantage of keeping the aspect ratio of scene images,
562 and the slight advantage of using dense crops. However, the
563 final results with People+Scene for the three methods don’t
564 have significant differences, this is due to the combination
565 with people information.

566 4.2. Convolutional Network feature analysis

567 **Yufei: Is this section necessary?** We use t-distributed
568 stochastic neighbor embedding technique [9] to visualize
569 the power of different layers’ features in convolutional net-
570 work. Two dimensional embedding of the high dimensional
571 CNN feature space is extracted, and we plot the data as 2-d
572 points with different colors indicating different classes they
573 belong to. Powerful features will pull data of different seman-
574 tic classes more apart.

575 We randomly choose one training-test partition and cor-
576 responding fine-tuned network parameters, and to avoid
577 overfitting effect, we examine the test set of Nets_{SDense} approach.
578 In Figure 3, each data is plotted as a dot in each figure.
579 The three columns correspond to data separation in first
580 layer, fourth layer and seventh layer respectively. The
581 first row visualizes features of all test data in scene CNN
582 CNN_{Scene}, and second row picks three classes from the
583 first row. The third row is features of CNN_{Person}, and the
584 three classes are picked for the last row. The three classes
585 chosen in second and last row is (goth:red, heavy:green,
586 surfer:blue).

587 In both CNN_{Scene} and CNN_{Person}, there is a clear trend
588 of class separation. As the layer ascends, , the data from
589 same class are more concentrate, and inter-class distance
590 are larger. The selected three classes show the trend more



591 Figure 3: Feature visualization of Nets_{SDense}: CNN_{Scene}
592 (first two rows) and CNN_{Person}(second two rows).
593 Rows: first: all scene images; second: all data from
594 goth/heavy/surfer urban tribes; third: all test person images;
595 fourth: all test data from goth/heavy/surfer urban tribes.
596 Columns: first: first convolutional layer; second:
597 fourth convolutional layer; third: seventh fully convolutional layer

598 clearly: they essentially form three clusters in the seventh
599 layer. As shown in the second and last rows, person data is
600 more challenging: The semantic classes are not as separate
601 as with the scene features.

602 4.3. Urban tribe classes vs. Imagenet classes

603 It’s recently being acknowledged that CNN pre-trained
604 features are generic and can be used for new tasks. In this
605 section we check the relationship between the new tasks and
606 original Imagenet task, using the urban tribe dataset, which
607 gives us some insight about the features extracted from pre-
608 trained network.

609 The urban tribe dataset contains groups of people, and
610 the important features for categorization are mainly human
611 related features, such as attire, make up, posture and expres-
612 sion.

648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701

sions. However, the ILSVRC dataset used for pretraining contains few images of humans. Instead of examining the output of the layers directly, we try to find the relationship between the 1000 classes in ILSVRC dataset and the classes in two image sets: urban tribe dataset, and person candidate images extracted from urban tribe dataset.

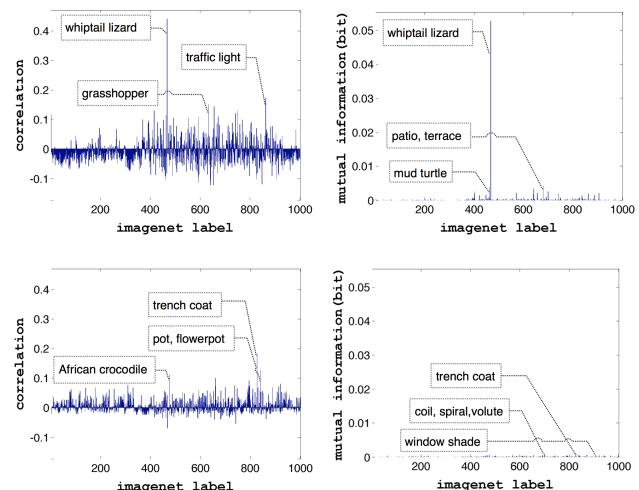
We use the parameters of pre-trained CNN network as our feature extraction model, and train a softmax layer on top of its 7th layer to predict the probabilities of one input image(either scene image or person candidate image) being in certain urban-tribe class $\Pr(l_{urban})$, where l_{urban} pre-trained is the 11 urban tribe categories. The output layer is trained for 3000 training iterations. We can also use the output of the pre-trained CNN network to predict probabilities of one input image being in certain Imagenet category, denoted as $\Pr(l_{Imagenet})$, where $l_{Imagenet}$ is the 1000 Imagenet categories. We use one round of 5-fold cross validation, and use all the images in urban tribe dataset for analysis.

We first check the relationshape of $\Pr(l_{urban})$ and $\Pr(l_{Imagenet})$ of person candidate images. We calculate the correlation coefficient of $\Pr(l_{urban})$ and $\Pr(l_{Imagenet})$ for all 1000 Imagenet classes, denoted as $R(\Pr(l_{urban}), \Pr(l_{Imagenet}))$. We also calculate the 1000 mutual information of the predicted score of urban tribe and Imagenet (where predicted score is 1 if the predicted label is the category being tested, 0 otherwise), denoted as $I(l_{urban}, l_{Imagenet})$.

In Figure 4, we choose two urban classes: *biker* and *hipster*, and plot the correlation coefficient R and mutual information I .The first row shows the result of *biker*, and second row *hipster*. For *biker*, there are several impulses in correlation plot, and one significant impulse in mutual information plot. *whiptail lizard* has high correlation with *biker*. Meanwhile, for *hipster*, which is the most difficult class, the correlation coefficient and mutual information are both low for all Imagenet classes.

To confirm the correlation,we use $\Pr(l_{Imagenet})$ directly as features, substituting the concatenated fc7fc6 features, and use the Nets_{SDistort} approach for classification. The accuracy is 52.68%. This decent result indicates the relationship between l_{urban} and $l_{Imagenet}$. Then, we check $l_{Imagenet}$ with highest correlation coefficients with l_{urban} . In Figure 5, we choose four l_{urban} : *formal*, *ravers*, *goth*, *hipster*, and choose some examples of person candidate images that have both high $\Pr(l_{Imagenet})$ and high $\Pr(l_{urban})$. We also show examples of the images in $l_{Imagenet}$. We can see some of the shared features between corresponding person images and Imagenet images, for example, similar shape for women tops and stingrays (Figure 5b).

There is a correlation between class-wise accuracy of predicted l_{urban} and the degree of relationship between predicted $\Pr(l_{urban})$ and $\Pr(l_{Imagenet})$, as shown in Figure 6.



702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755

Figure 4: Correlation and mutual information of l_{urban} and 1000 $l_{Imagenet}$. The first row is for $l_{urban}=\text{Biker}$. The second row is for $l_{urban}=\text{Hipster}$. Top three Imagenet classes are marked.

Class-wise accuracy is calculated for person candidate images(Figure 6a, 6b) or scene images(Figure 6c, 6d). For each l_{urban} , the maximum correlation/mutual information over 1000 Imagenet classes are used to indicate the degree of its relationship with $l_{Imagenet}$.

The correlation between Imagenet class and urban tribe class and its relationship with class-wise recognition rate may indicate that the “generic” features extracted by pre-trained CNN networks are not so generic. The network is trained to separate the Imagenet classes most, and if we use the features for a new classification task, the performance of the task is related to how well the new classes can be “mapped” to the Imagenet classes.

5. Conclusion

In this work, we explore the use of pre-trained deep network features for social group recognition. We propose a recognition framework which takes in both individual and global features. Our results show the success of our framework. Both visualization and numeric results show the generalization ability and semantic information of pre-trained CNN features. Our results also show the necessity of fine-tuning in social group recognition and imply the potential usage and modification of pre-trained CNN features for other computer vision tasks.

References

- [1] A. Dhall, J. Joshi, I. Radwan, and R. Goecke. Finding happiest moments in a social context. In K. M. Lee, Y. Matsushita,

756

757

758

759

760

761

762

763

764

765

766

767

768

769

770

771

772

773

774

775

776

777

778

779

780

781

782

783

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

806

807

808

809

810

811

812

813

814

815

816

817

818

819

820

821

822

823

824

825

826

827

828

829

830

831

832

833

834

835

836

837

838

839

840

841

842

843

844

845

846

847

848

849

850

851

852

853

854

855

856

857

858

859

860

861

862

863

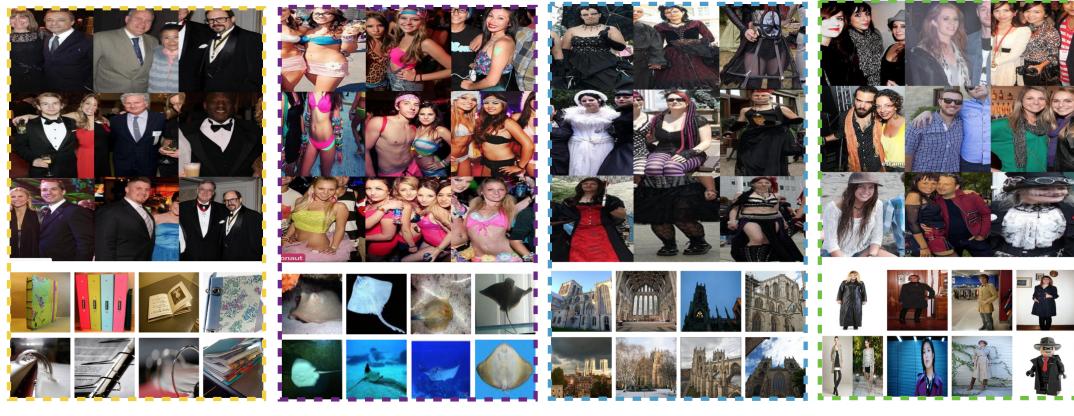


Figure 5: selected urban tribe classes and the corresponding highest correlation $l_{Imagenet}$. Upper nine images: person candidate images with high $\text{Pr}(l_{Imagenet})$ and high $\text{Pr}(l_{urban})$. Lower eight images: example of images in $l_{Imagenet}$.

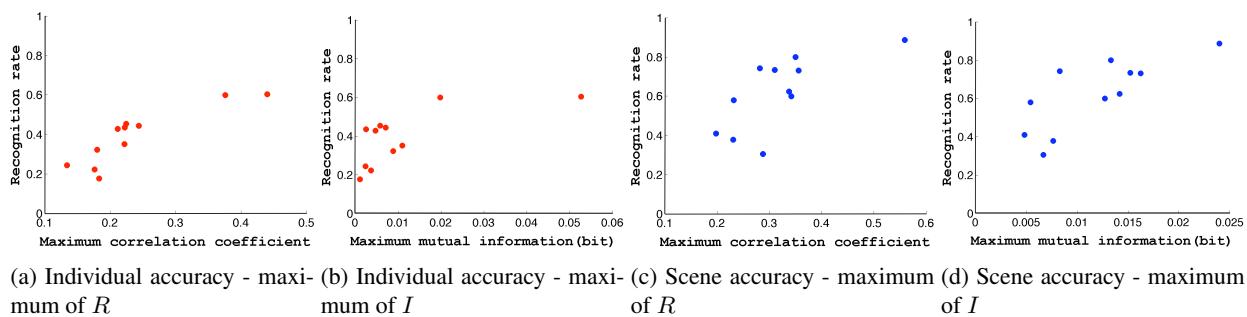


Figure 6: The relationship between class-wise recognition rate and maximum of correlation $R(\text{Pr}(l_{urban}), \text{Pr}(l_{Imagenet}))$, class-wise recognition rate and maximum of mutual information $I(l_{urban}, l_{Imagenet})$

- J. M. Rehg, and Z. Hu, editors, *ACCV 2*, volume 7725, pages 613–626. Springer, 2012.
- [2] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *CoRR*, abs/1310.1531, 2013.
- [3] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.
- [4] A. C. Gallagher and T. Chen. Understanding images of groups of people. In *CVPR*, pages 256–263. IEEE, 2009.
- [5] P.-H. Gosselin, N. Murray, H. Jégou, and F. Perronnin. Inria+xerox@fgcomp: Boosting the fisher vector for fine-grained classification. Technical Report 0, INRIA, December 2013.
- [6] Y. Jia. Caffe: An open source convolutional architecture for fast feature embedding. <http://caffe.berkeleyvision.org/>, 2013.
- [7] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In

- P. L. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *NIPS*, pages 1106–1114, 2012.
- [8] I. S. Kwak, A. C. Murillo, P. Belhumeur, S. Belongie, and D. Kriegman. From bikers to surfers: Visual recognition of urban tribes. In *British Machine Vision Conference (BMVC)*, Bristol, September 2013.
- [9] G. H. Laurens van der Maaten. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, November 2008.
- [10] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1:541–551, 1989.
- [11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, volume 86, pages 2278–2324, 1998.
- [12] A. C. Murillo, I. S. Kwak, L. Bourdev, D. Kriegman, and S. Belongie. Urban tribes: Analyzing group photos from a social perspective. In *CVPR Workshop on Socially Intelligent Surveillance and Monitoring (SISM)*, Providence, RI, June 2012.

864	[13] G. Wang, A. C. Gallagher, J. Luo, and D. A. Forsyth. Seeing	918
865	people in social context: Recognizing people and social rela-	919
866	tionships. In K. Daniilidis, P. Maragos, and N. Paragios, ed-	920
867	itors, <i>ECCV</i> , volume 6215, pages 169–182. Springer, 2010.	921
868	[14] M. D. Zeiler and R. Fergus. Visualizing and understanding	922
869	convolutional networks. <i>CoRR</i> , abs/1311.2901, 2013.	923
870		924
871		925
872		926
873		927
874		928
875		929
876		930
877		931
878		932
879		933
880		934
881		935
882		936
883		937
884		938
885		939
886		940
887		941
888		942
889		943
890		944
891		945
892		946
893		947
894		948
895		949
896		950
897		951
898		952
899		953
900		954
901		955
902		956
903		957
904		958
905		959
906		960
907		961
908		962
909		963
910		964
911		965
912		966
913		967
914		968
915		969
916		970
917		971