# 基于图像三元组挖掘的无监督视觉表示学习

何果财 刘峡壁

(北京理工大学计算机学院智能信息技术北京市重点实验室 北京 100081)

**摘 要** 特征表示是计算机视觉应用中的关键问题之一,由于视觉数据的海量增长和人工标记的高成本问题,无监督视觉表示学习逐渐受到了人们的广泛关注.该文提出一种基于图像三元组挖掘的无监督视觉表示学习方法.该方法包含两个部分:挖掘图像三元组和学习特征表示.具体而言,首先构建二分类卷积神经网络,从图像数据集中挖掘三元组,即图像、与其相似的图像、与其不相似的图像这三者构成的数据.然后使用得到的图像三元组作为监督信息,通过训练 Triplet 卷积神经网络来获得视觉表示.为了验证该文所提出的学习方法,该文将学习得到的特征表示应用到常用图像数据集的聚类和分类问题上.在聚类方面,相较于传统的视觉特征方法,该文特征的聚类表现在规范化互信息上有 12.7%的提升;在分类方面,所提出的方法也取得了有竞争力的结果,实验表明该文所提出的方法是有效的.

**关键词** 无监督学习;视觉表示学习;图像三元组;卷积神经网络;深度学习

**中图法分类号** TP183 **DOI 号** 10.11897/SP.J.1016.2018.02787

# Unsupervised Visual Representation Learning with Image Triplets Mining

HE Guo-Cai LIU Xia-Bi

(*Beijing Laboratory of Intelligent Information, School of Computer Science, Beijing Institute of Technology, Beijing* 100081)

**Abstract** Feature representation is one of the key problems in the field of computer vision, good feature representation can improve the performance of machine learning algorithms. Deep learning is one of the best methods of learning visual representation at present. The supervised method can provide rich features for classification and recognition algorithms. However, due to the massive growth of visual data and the high cost of manual annotation, unsupervised learning of visual representation has gradually received more attentions. This paper presents an unsupervised deep learning method based on image triplets mining for learning visual representation of images. Our method consists of two stages: mining image triplets and learning feature representation of images. Specifically, first, we constructed a convolutional neural network (CNN) for binary classification, then we sampled data from original datasets for the binary classification. The first class of data was obtained by data augmentation. We performed a series of visual transformation to an image and yielded some images which made up the first class of data. And we randomly sampled data from the remaining images as the second class of data. We used these two class of data to train a CNN and utilized the characteristic of soft-max activation function to mine a large number of image triplets from original image dataset. An image triplet consists of an image, an image which is similar to it and an image which is dissimilar to it. Second, we designed a Triplet CNN, which consisted of three channels of CNN and the three channels shared parameters. And then we fed the image triplet samples into the Triplet CNN. These image triplets can provide

收稿日期:2017-07-02;在线出版日期:2018-03-26.本课题得到国家自然科学基金(60973059,81171407)、教育部新世纪优秀人才支持计划(NCET-10-0044)资助.**何果财**,男,1993 年生,硕士研究生,主要研究方向为机器学习、计算机视觉.E-mail:heguocai@bit.edu.cn.
**刘峡壁**,男,1972 年生,博士,副教授,中国计算机学会(CCF)高级会员,主要研究领域为机器学习、模式识别、计算机视觉、信息检索.

supervisory information to Triplet CNN for representation learning. We used the appropriate triplet loss to optimize the Triplet model. After the completion of training of the Triplet CNN, we input all images of the original dataset to the Triplet model and can obtain the visual representations of all images from the dataset. In the entire algorithm process, our method definitely exploited no annotation information. In order to evaluate the proposed method, we applied the feature representations learned by our method to the applications of clustering and classification on the commonly used image datasets. In the clustering tasks on multiple image datasets, the effect of the learned representation on benchmark clustering algorithms is averagely up to 15.3% in normalized mutual information (NMI), and compared with the traditional visual feature mining method, the performance of the proposed method has also achieved an improvement of about 12.7% in NMI. For the classification, based on the learned feature representation, we just used shallow classifiers. We still obtained competitive performance when compared with the best classification results on several benchmark datasets, and in another part of benchmark datasets, we got the best results that we know so far. According to the visualization results of the features of several datasets, we can also see that the feature representation learned by our method have good discriminability. The results of experiments convincingly demonstrate that the method we proposed is effective.

**Keywords** unsupervised learning; visual representation learning; image triplets; convolutional neural networks; deep learning



## 1 引 言

视觉表示是计算机视觉应用中的基础问题之一.在深度学习引起人们的广泛关注之前,计算机视觉研究主要依赖于人为设计来获得有效的特征,例如 HOG[1]、SIFT[2]、LBP[3]等广为人知的图像特征提取方法.这些传统方法均依赖于专家知识,不具备广泛的适用性,在处理不同类型的视觉任务时,通常需要采用不同的特征.近年来,随着深度学习[4]在计算机视觉领域的成功应用,采用深度神经网络从大量视觉数据中自动学习特征表示逐渐得到了人们的广泛关注.其中,卷积神经网络[5]在图像与视频领域应用尤为广泛.通常我们将用于分类的卷积神经网络的最顶层视为分类器,将其它层视为特征表示提取器,其它层直接从图像像素数据中学习特征表示,再将提取得到的表示输入到顶层分类器中完成分类.在卷积神经网络学习框架中,其参数通常采用梯度下降和反向传播方法,在分类数据集上进行训练.因此对于特征提取而言,这是监督的学习模式,需要大量的人工标记数据.然而现实情况是,由于人工标记的高成本问题,视觉数据虽呈爆炸式增长,带标记数据却只占很少的一部分,这在一定程度上制约了监督学习的应用.如果可以利用无标记的视觉数据去学习有效的特征表示,便可以大大减少对人工标记的依赖,从而使大量的无标记数据得到利用,是一种具有发展前景的解决图像表示问题的途径.

本文提出一种基于图像三元组挖掘的无监督视觉表示学习方法.图 1 描述了本文方法的总体框架,该方法分为两部分,首先基于二分类卷积神经网络挖掘图像数据集中的三元组($p$,$p^+$,$p^-$),其中 $p$ 表示数据集中任意一张图像,$p^+$ 表示与 $p$ 相似的图像,$p^-$ 表示与 $p$ 不相似的图像,然后使用挖掘得到的图像三元组训练 Triplet 深度神经网络,获得具有判别性的特征表示.

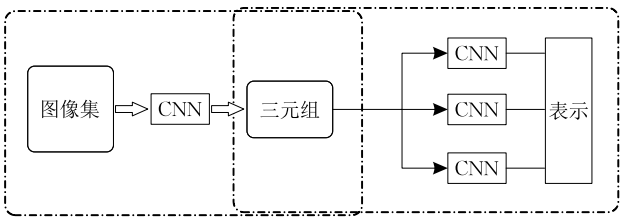



图 1 本文方法总体框架

我们的学习目标是相似图像在表示空间中应具有相似的特征表示,不相似图像在表示空间中应具有相异的特征表示.因此试图缩小相似图像对($p$,$p^+$)的特征表示之间的距离,同时增大不相似图

像对$(p,p^-)$的特征表示之间的距离，从而使学习得到的表示空间具备更好的判别性.需要说明的是，图像相似性通常具有不同的定义，从不同的角度可以定义不同的相似性.本文从类别语义角度来定义图像的相似性，即图像属于相同类别，则认为其是相似的，否则是不相似的.

我们分别在 MNIST、CIFAR-10、COIL-20 等常用数据集上进行了实验.一方面对比分析了挖掘出的图像三元组的质量，另一方面将学习得到的特征表示应用于分类和聚类问题，以检验其是否具有良好的判别性.

总结下来，本文的主要贡献是：

(1)提出了一种基于卷积神经网络的，在无标记图像数据集中挖掘图像三元组的方法.实验结果表明该方法挖掘的图像三元组具有较高的准确率，且包含更多的难三元组样本，可为后续表示学习提供有效的监督信号.

(2)将挖掘得到的图像三元组用于训练深度卷积网络，以学习图像的特征表示.分类和聚类的实验结果表明，本文提出的方法学得的表示具有良好的判别性.

## 2　相关工作

本文主要在基于深度神经网络的视觉表示学习和无标记视觉数据挖掘上进行了探索，下面分别介绍前人在这两方面的相关工作.

### 2.1　视觉表示学习

我们将基于神经网络的视觉表示学习方法分为监督学习和无监督学习两类.

在深度有监督视觉表示学习方面，文献[6-7]中综述了深度前馈卷积神经网络，可学习从低层到高层的层次化的特征表示，神经网络的较低层学得的特征属于低级视觉，较高层学得的表示是高级视觉.Triplet/Siamese 神经网络[8-9]常用于度量学习和表示学习中，将图像数据映射到低维的表示空间中，同一类别的图像在表示空间的距离上比在原空间更加接近，而不同类别的图像在表示空间的距离上比原空间更远.如 Deep Ranking[10]引入深度 Triplet 神经网络学习自然图像的细粒度相似性；Deep Hashing[11]基于 Triplet 框架学习图像的深度哈希，并用于图像检索；文献[12]中提出 DDML 结构，利用深度 Siamese 网络结构解决人脸验证问题.以上监督式学习方法需要大量带标记的图像数据进行训练.

深度无监督视觉表示学习[13-18]方面，目前方法主要包含三类：基于重建输入的方法、基于无标记数据挖掘的方法、基于特定任务的方法.第一类以自编码器[15] Auto-Encoder(AE)为代表，它试图学习输入数据中的主要特征，利用神经网络的高度非线性特性，学习比非线性主成分分析更有效的特征表示.在此基础上，研究者们进行了很多扩展.稀疏自编码器[19](Sparse Auto-Encoder)为损失函数引入稀疏惩罚项，更适于为分类问题学习特征表示.降噪自编码器[20](Denoising Auto-Encoder)引入随机噪声，使得网络不仅能重建输入，同时具备去除噪声的能力，可学习更加鲁棒的特征.相比于浅层的自编码器，深度自编码器[21](Deep Auto-Encoder)可以获得对数据更好的压缩表示.卷积自编码器[22](Convolutional Auto-Encoder)引入了卷积运算，使模型更关注图像的局部特征.然而，以上基于重建输入的方法均未将特征表示的判别性作为学习目标，不利于学习具有判别性的视觉特征.

第二类基于无标记数据挖掘的方法从无标记数据集中挖掘各种类型的重要信息作为监督数据，在此基础上使用深度结构学习特征表示.文献[23]提出利用无监督视频跟踪方法，挖掘相似的图像块作为后续训练的监督信息.文献[24]将图像进行分块，将每个图像块之间的相对位置信息作为监督信息.文献[25]基于测地线距离挖掘相似和不相似图像对作为监督信息.文献[26]和文献[27]都基于视频信息学习视觉表示，分别利用视频的时序信息和第一视角视频中的自运动信息作为监督信息，并以此训练深度神经网络以获得视觉表示.然而，文献[23,26-27]中提出的表示学习方法不是以静态图像数据作为研究对象；文献[24]学习图像块之间的相对位置特征，而不是以学习具有判别性特征为目标；基于图模型[25]采用 $K$-近邻图的测地线距离度量图像相似性，而模型中近邻图的构建方法仍以欧式距离为基础，且近邻图的 $K$ 值通常对模型的性能影响较大，导致模型不够鲁棒.

第三类基于特定任务的方法将表示学习与应用目标融合在一起.文献[18]将 Deep Auto-Encoder 与基于 Kullback-Leibler(KL)散度聚类相结合，通过最小化当前弱聚类中心的概率分布与辅助目标概率分布之间的 KL 散度，来优化深度自编码器特征表示和聚类结果；文献[28]中构建了深度卷积表示

和图像聚类联合学习的框架，即图像表示可以提高聚类结果，反之利用聚类结果可以进一步优化图像的特征表示，从而形成一个循环自提升的过程.

本文所提出的方法属于上述第二类基于无标记数据挖掘的视觉表示学习方法. 与文献[23,25]相似，我们采用无监督学习方法挖掘图像三元组，并使用 Triplet 深度卷积神经网络学习视觉表示. 与文献[23]不同的是，我们提出的图像三元组挖掘方法针对的是静态图像数据集，而文献[23]的学习数据源于视频；与文献[25]不同的是，我们提出的图像三元组挖掘方法基于卷积神经网络，相比而言，文献[25]中基于距离度量来挖掘图像三元组，而本文是通过学习的方式来挖掘图像三元组.

另外，本文提出的方法是完全无监督的. 我们在训练神经网络时，网络的初始参数没有采用文献[29]中所使用的参数预训练方法，在 ImageNet 等大型分类数据集上进行参数预训练实际是一种半监督的学习方法，尽管在目标数据集上没有使用标签信息，但在预训练阶段却使用到了其它数据集的监督信息.

### 2.2 无标记视觉数据挖掘

应用 Triplet/Siamese 神经网络，通常情况下，需要提供 Triplet/Siamese 样本对网络进行训练. 对于监督学习模式的 Triplet/Siamese 应用[10-12]，根据已有的图像标签可以很容易地获取 Triplet/Siamese 样本，相似的图像对取自相同类别下的图像，不相似的图像对取自不同类别下的图像即可. 而对于无监督学习模式而言，则需要设计可自动地从无标记视觉数据中挖掘图像对的算法. 文献[23]中提出的图像对挖掘方法中，作者认为在不同视频帧中某个被跟踪对象所在图像块之间是相似的，而来自于不同视频中随机采样得到的图像块之间是不相似的. 文献[26]利用视频蕴含的时序信息，获取视频帧时序三元组，在此基础上学习深度表示，用于检验来自一个视频中若干视频帧是否为正确的时序. 文献[24]利用图像块的相对位置关系组成 Siamese 样本，并作为监督信号训练深度网络. 文献[25]是使用图像间的欧式距离建立 $K$-近邻图，进而应用最短路径算法，基于测地线距离挖掘相似和不相似图像对. 该模型在构建近邻图时，需要进行精细的参数选择，若 $K$ 值较大，则距离较远的节点可能被误认为近邻，出现“短路”问题；若 $K$ 值较小，则近邻图中某些区域可能被孤立，出现“断路”问题. 此外，新样本的加入可能引起近邻图的结构发生改变，影响模型效率.

本文提出一种基于二分类卷积神经网络挖掘图像三元组的方法. 与文献[16-17]采用的方法相似，一张图像经过一系列不同类型的图像变换操作得到若干张相似的图像. 但与之不同的是，首先，我们的方法提供了更多的变换方式；其次，引入图像变换的目的不仅是学习不变性特征，而且要从图像数据集中挖掘图像三元组；另外，本文所提出的图像三元组挖掘方法是一种基于学习的方法，相比文献[23,25]提出的挖掘方法更具有适应性，神经网络的强学习能力使得我们的方法相对更易于应用到多种图像数据集.

## 3 图像三元组挖掘方法

本节提出基于二分类神经网络挖掘图像三元组的方法. 图 2 描述了该方法的总体框架. 首先，随机地从图像集中选择一张图像 $p$，应用图像变换可生成一系列与图像 $p$ 相似的图像集合 $\boldsymbol{T}_p$，然后从图像集合中除图像 $p$ 之外的图像中随机采样得到图像子集 $\boldsymbol{T}_r$. 将 $\boldsymbol{T}_p$ 和 $\boldsymbol{T}_r$ 两个图像集标定为不同的类别，

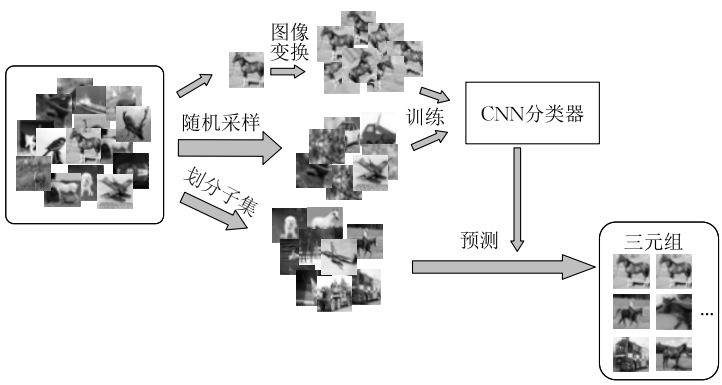



图 2 本文图像三元组挖掘方法示意图

进而利用它们训练二分类卷积网络.接下来,从原图像数据集中随机选择部分图像作为预测图像集,预测图像集合不包含 $\boldsymbol{T}_r \cup \{p\}$ 集合中的图像.从预测图像集合中找到与图像 $p$ 相似的图像和不相似的图像,得到图像三元组.这种在随机划分子集而不是整个数据集上进行预测的方式,可以帮助挖掘出更为鲁棒的图像三元组,同时也提高了算法的效率.

下面分为两部分对上述方法进行详细介绍,第 3.1 节介绍图像生成模块,第 3.2 节描述在此基础上的三元组挖掘算法.

### 3.1　图像生成

在提出的图像三元组挖掘方法中,我们利用图像生成技术得到大量图像并用于后续计算.实际上,对于图像集中任意一张图像 $p$,使用以下图像变换操作可以得到大量与该图像相似的图像.本文采用的图像变换操作包括:旋转、平移、缩放、水平镜像、剪切、高斯模糊、色调微调.

(1)旋转.将图像按顺时针或逆时针方向进行旋转,限制最大旋转角度.

(2)平移.对图像进行垂直和水平方向平移.

(3)缩放变换.在一定范围内将图像放大或缩小.

(4)水平镜像.对图像进行左右翻转,不使用上下翻转.

(5)剪切变换.对图像进行拉伸.

(6)高斯模糊.达到降噪的目的.

(7)色调微调.在图像 HSV 空间微调 H 通道的值,图像的所有像素点的微调幅度一致.

每生成一幅图像时,选取以上至少一种图像变换操作进行处理.处理过程中,随机化所选择的图像变换操作的处理顺序,即各操作之间没有特定的顺序,是完全随机的;另外各变换操作的参数值也在限定范围内根据均匀分布随机选取.这样,根据单个种子图像 $p$,利用有限的图像变换操作即可以生成大量的图像.为了避免引入不必要的噪声,针对不同的目标数据集,我们选择的图像变换操作集合和各图像变换操作的参数取值范围不完全相同.例如,在 CIFAR-10 数据集上我们使用色调微调的变换操作,而在 COIL-100 中同类图像间没有颜色的变化,使用色调微调的变换则会引入不必要的噪声.具体分析请见实验部分.

通过对种子图像 $p$ 应用多次图像变换,生成与之相似图像集合 $\boldsymbol{T}_p$,将 $\boldsymbol{T}_p$ 与图像 $p$ 标定为同一类图像.与图像数据集中图像 $p$ 所属真实类别不同,此处标定的类别没有特定语义,仅表示图像集 $\boldsymbol{T}_p$ 与图像 $p$ 相似.下面我们将利用生成的该图像集合,基于卷积神经网络学习图像 $p$ 所在类别的不变性特征,用于从图像集中挖掘图像三元组.

### 3.2　挖掘方法

给定无标注图像集 $\boldsymbol{I}$,从 $\boldsymbol{I}$ 中任意挑选一张图片 $p$.我们将针对 $p$,从 $\boldsymbol{I}$ 中挖掘得到若干 $p^+$ 和 $p^-$.首先,多次应用 3.1 节中所描述的图像生成方法,可得到与图像 $p$ 相似的图像集合 $\boldsymbol{T}_p$,将 $\boldsymbol{T}_p$ 集合中的图像与图像 $p$ 视为同一类图像,将图像 $p$ 并入 $\boldsymbol{T}_p$ 中,$\boldsymbol{T}_p = \boldsymbol{T}_p \cup \{p\}$.接下来,我们从图像集合 $\{\boldsymbol{I} - p\}$ 中,随机选择若干张图像构成一个图像集合,用 $\boldsymbol{T}_r$ 表示.可认为图像集合 $\boldsymbol{T}_r$ 中的图像与图像 $p$ 的类别不相同,则 $\boldsymbol{T}_p$ 和 $\boldsymbol{T}_r$ 两个集合类别不同.我们赋予 $\boldsymbol{T}_p$ 和 $\boldsymbol{T}_r$ 两个集合不同的标签,令 $\boldsymbol{T}_p$ 集合的标签为 0,$\boldsymbol{T}_r$ 集合的标签为 1,$\boldsymbol{T}_p \cup \boldsymbol{T}_r$ 可作为二分类任务的训练数据,在此基础上我们学习得到一个相应的二分类神经网络.由于 $\boldsymbol{T}_p$ 中的图像之间具有较强的相关性,而且随机挑选的 $\boldsymbol{T}_r$ 中可能有少部分图像与 $\boldsymbol{T}_p$ 属于同一类别.在这种情况下,我们设计结构简单的神经网络来学习 $\boldsymbol{T}_p$ 与 $\boldsymbol{T}_r$ 集合上的不变性特征,以尽可能地减少过拟合的可能,保证对数据集 $\boldsymbol{T}_p \cup \boldsymbol{T}_r$ 有较好的学习能力.具体地,我们使用卷积神经网络作为二分类模型,输出层含两个神经元,使用 soft-max 作为激活函数:

$$\boldsymbol{a}_j^L = \frac{\mathrm{e}^{\boldsymbol{z}_j^L}}{\sum_k \mathrm{e}^{\boldsymbol{z}_k^L}} \tag{1}$$

其中,$\boldsymbol{z}_j^L$ 表示输出层(第 $L$ 层)第 $j$ 个神经元激活前的值,$\boldsymbol{a}_j^L$ 表示输出层的第 $j$ 个神经元的激活值.显然有 $\sum_j \boldsymbol{a}_j^L = 1$,因此输出层的第 $j$ 个神经元的激活值表示输入样本属于该神经元对应第 $j$ 类别的估计概率值.除输出层以外的各层,都使用线性矫正单元(*relu*)作为激活函数,可表示为

$$relu(x) = \max\{0, x\} \tag{2}$$

二分类网络的训练数据为 $\boldsymbol{I}_{\text{train}} = \boldsymbol{T}_p \cup \boldsymbol{T}_r$,$\boldsymbol{T}_p$ 对应的期望输出为标签 0 的 one-hot 编码[1,0],$\boldsymbol{T}_r$ 对应的期望输出为向量[0,1].

本文使用交叉熵作为分类损失函数.设 $\boldsymbol{S}$ 为输出层 soft-max 函数的激活值,设 $\mathcal{L}$ 为预期输出,则交叉熵为

$$CE(S, \mathcal{L}) = -\sum_i \mathcal{L}_i \log(s_i) \tag{3}$$

需要注意的是,由于训练样本较少且有一定的错误样本,因此除了构建结构简单的神经网络外,我们进

一步使用 $L_2$ 正则、Dropout 方法进一步减少过拟合的影响. 首先,$L_2$ 正则化方法在模型损失函数中增加正则项,可表示为

$$\mathbb{C}=\mathbb{C}_0+\frac{\lambda}{2n}\sum_w w^2 \tag{4}$$

其中,$\mathbb{C}_0$ 表示原始未正则化的损失函数,损失函数值与正则化程度之间的相对重要性由正则化常数 $\lambda$ 进行折中. 在模型优化的过程中,正则化要求降低神经网络的权值,以防止出现过大的权值导致过拟合. 其次,通过 Dropout 方法,在训练过程中修改网络结构,做进一步的正则化:

$$\text{Dropout}(x)=\begin{cases}\dfrac{x}{kp}\times mask, & \text{训练阶段}\\ x, & \text{测试阶段}\end{cases} \tag{5}$$

在神经网络训练阶段,$kp$ 表示 Dropout 层的输入 $x$ 中每个元素被保留的概率值,取值为(0,1]. 根据 $kp$ 参数随机生成与输入 $x$ 具有相同维度的张量 $mask$,张量中元素取值 0 时表示 $x$ 对应元素在单次训练中被弃用,取值 1 则表示对应元素被保留. 在测试阶段,Dropout 层直接传递输入 $x$,不做实质性操作. 这种神经网络集成的设计模式,可以帮助减少同层神经元之间过度依赖的关系,从而增加模型的泛化能力.

在训练集 $\boldsymbol{I}_{\text{train}}$ 上训练完成之后,我们随机地从图像集合$\{\boldsymbol{I}-\boldsymbol{T}_r-p\}$中抽取图像子集 $\boldsymbol{I}_{\text{pred}}$. 使用训练完成的二分类神经网络对图像集合 $\boldsymbol{I}_{\text{pred}}$ 进行预测,经过 soft-max 层激活得到预测输出 $\boldsymbol{S}_{\text{pred}}$. 要进一步得到与图像 $p$ 相似的图像,只需找到 $\boldsymbol{S}_{\text{pred}}$ 中第一维值最大者,它对应在子集 $\boldsymbol{I}_{\text{pred}}$ 中的图像与图像 $p$ 相似. 要得到不相似的图像,只需找到 $\boldsymbol{S}_{\text{pred}}$ 中第二维值最大者,它对应在子集 $\boldsymbol{I}_{\text{pred}}$ 中的图像与图像 $p$ 不相似.

$$p^+=\arg\max_{p'}\boldsymbol{S}^{:,0}_{\text{pred}} \tag{6}$$

$$p^-=\arg\max_{p'}\boldsymbol{S}^{:,1}_{\text{pred}} \tag{7}$$

式(6)和式(7)中 $p'\in\boldsymbol{I}_{\text{pred}}$,$\boldsymbol{S}^{:,0}_{\text{pred}}$ 表示 $\boldsymbol{I}_{\text{pred}}$ 集合中所有图像通过神经网络输出张量的第一维,$\boldsymbol{S}^{:,1}_{\text{pred}}$ 表示 $\boldsymbol{I}_{\text{pred}}$ 集合中所有图像通过神经网络输出张量的第二维. 至此,得到了一个图像三元组样本$(p,p^+,p^-)$.

为了提高算法的效率,我们增加以下两个技巧:(1)训练一次神经网络,重复 $m$ 次随机采样得到 $m$ 个 $\boldsymbol{I}_{\text{pred}}$,进行 $m$ 次预测挖掘得到 $m$ 个图像三元组样本$(p,p^+,p^-)$;(2)本文定义的相似性是类别语义相似性,因此利用直递性可快速扩充三元组的数量. 设有三元组$(p,p^{+(1)},p^{-(1)})$和$(p,p^{+(2)},p^{-(2)})$,则图像 $p^{+(1)}$ 与 $p^{+(2)}$ 也是相似的,即可以进一步得到三元组$(p^{+(1)},p^{+(2)},p^{-(1)})$和$(p^{+(1)},p^{+(2)},p^{-(2)})$. 上述方法,从图像集合中挖掘出了对应于图像 $p$ 的图像三元组. 为了挖掘更多的三元组,则继续随机挑选种子图像 $p$,重复上述算法即可. 这一过程总结如算法 1 所示.

**算法 1.** 无监督图像三元组挖掘算法.

输入:无标记图像数据集 $\boldsymbol{I}$,需要挖掘的种子图像个数 $L$,一个种子图像对应的两个自定义类集合的图像个数 $N_1$ 和 $N_2$,随机采样预测子集大小 $M$,对每个种子图像迭代挖掘次数 $m$.

输出:图像三元组集合 $\boldsymbol{H}$

```
种子图像集合 seeds={}
FOR l=1 TO L
    随机挑选种子图像 p
    WHILE (p in seeds)
        随机挑选种子图像 p
    END WHILE
    seeds=seeds∪{p}
    根据种子图像 p 通过图像变换生成与其相似的图像集合 T_p, |T_p|=N_1,标签为 0
    T_p=T_p∪{p}
    随机选取数据集 I 中排除图像 p 的图像子集 T_r, |T_r|=N_2,标签为 1,实验中选取 N_1=N_2
    在训练数据集 I_train=T_p∪T_r 上训练二分类 CNN
    以图像 p 为种子图像的三元组集合 H_p=∅
    FOR i=1 TO m
        随机选取数据集 I 中排除 T_r∪{p}的图像子集 I_pred, |I_pred|=M
        使用训练好的 CNN 模型预测 I_pred 得到 S_pred
        应用式(6)和式(7)得到 p+ 和 p-,得到三元组 (p,p+,p-)
        H_p=H_p∪{(p,p+,p-)}
    END FOR
    利用直递性快速扩充得到三元组集合 H_aug
    H=H∪H_p∪H_aug
END FOR
```

## 4 学习视觉表示

利用上一节介绍的方法挖掘得到图像三元组后,本节基于 Triplet 神经网络,从图像三元组中学习视觉表示.

### 4.1 Triplet 卷积神经网络结构

由于图像类内的差异性和类间的相似性,在图像原始空间和不合适的距离度量下,类内图像之间的距离可能反而比不同类别图像之间的距离大,此种情况对于视觉算法是不利的. 本文以学习具有良好判别性的特征表示为目标,使得在学习得到的表

示空间中，相似图像之间的距离更近，而不相似的图像之间的距离尽可能地远. 我们在图 3 中展示了这样的学习目标.

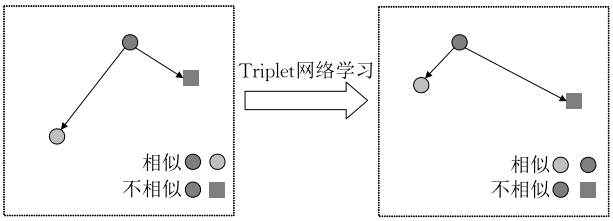



图 3　Triplet 卷积神经网络学习目标

为了利用上一节挖掘得到的图像三元组来学习具有判别性的视觉表示，我们设计了一个三通道的深度 Triplet 卷积神经网络，每个通道都是基本的卷积神经网络，且三个通道共享网络参数. 该结构如图 4 所示. 其中，三个通道各自以三元组的各项为输入，每个通道中堆叠了卷积层和全连接层，三个通道共享的网络参数采用误差逆传播算法进行同步更新. 从参数个数的角度看，共享全部参数的三通道网络结构与单通道结构的参数数量是相同的，但结合三元组和 Triplet 损失函数的指导学习之后，模型所学的特征表示与分类网络的特征表示是不相同的；通过共享三个通道的全部参数可降低整个模型的参数数量，加快网络的学习过程，并降低过拟合风险；另外，相比于单通道结构结合 Triplet 损失函数，三通道的结构可并行计算，大大减少了对计算资源和时间的消耗.

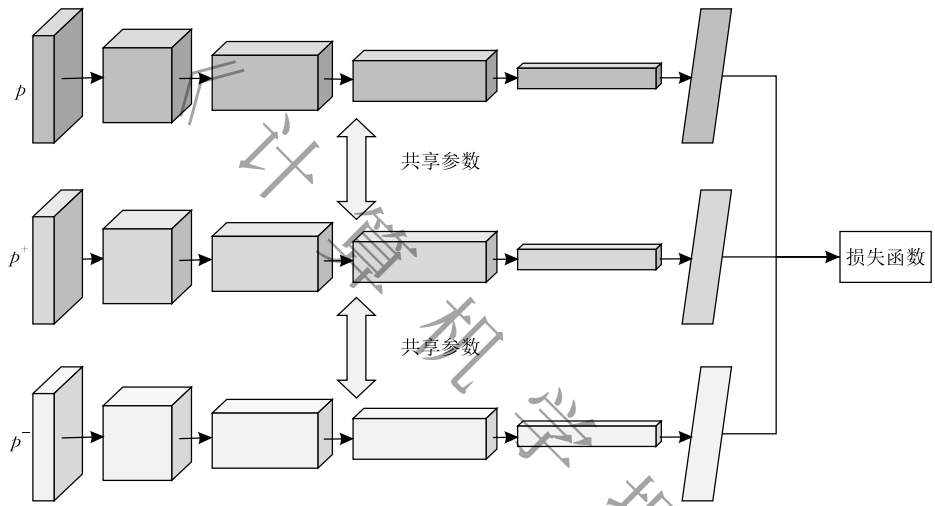



图 4　本文提出的 Triplet 神经网络结构

### 4.2　学习准则

根据上述学习目标，我们设计了 Triplet 神经网络的损失函数. 设 $f(\cdot)$ 表示 Triplet 神经网络表示的非线性映射，则图像三元组 $(p,p^+,p^-)$ 在表示空间中为 $(f(p),f(p^+),f(p^-))$，使用欧式距离度量图像在表示空间中的距离：

$$\begin{cases} D(p,p^+)=\|f(p)-f(p^+)\|_2 \\ D(p,p^-)=\|f(p)-f(p^-)\|_2 \end{cases} \tag{8}$$

对于图像三元组的集合 $\boldsymbol{H}$，用 $\boldsymbol{H}^{(i)}$ 表示第 $i$ 个三元组 $(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_2^{(i)},\boldsymbol{H}_3^{(i)})$，则在特征空间中，正样本之间的距离表示为 $\mathcal{D}(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_2^{(i)})$，负样本之间的距离表示为 $\mathcal{D}(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_3^{(i)})$. 根据本文提出的表示学习目标，我们应使得 $\mathcal{D}(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_2^{(i)})<\mathcal{D}(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_3^{(i)})$，因此我们将损失定义为

$$\mathbb{L}(\boldsymbol{H}^{(i)})=\frac{\mathcal{D}(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_2^{(i)})}{\mathcal{D}(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_3^{(i)})+C} \tag{9}$$

其中 $C$ 是修正常数，防止分母为 0，实验中取 $C=0.0001$. 显然 $\mathbb{L}(\boldsymbol{H}^{(i)})\geq 0$. 我们要在最小化 $\mathcal{D}(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_2^{(i)})$ 的同时最大化 $\mathcal{D}(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_3^{(i)})$，即使得 $\mathbb{L}(\boldsymbol{H}^{(i)})$ 尽可能的小. 因此我们定义 Triplet 神经网络的目标函数为

$$\min_{\boldsymbol{W},\boldsymbol{b}}\sum_i\frac{\mathcal{D}(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_2^{(i)})}{\mathcal{D}(\boldsymbol{H}_1^{(i)},\boldsymbol{H}_3^{(i)})+C}+\frac{1}{2}\lambda\|\boldsymbol{W}\|_2^2 \tag{10}$$

其中，$\boldsymbol{W}$ 和 $\boldsymbol{b}$ 分别是网络的权值和阈值参数，$\frac{1}{2}\lambda\|\boldsymbol{W}\|_2^2$ 是 $L_2$ 正则化. 引入正则项可避免在神经网络的训练过程中产生过大的权值，减少训练过程的过拟合问题.

基于式(10)，采用随机梯度下降的方法，对上述 Triplet 神经网络进行训练. 在训练完成后，将原图像集中的图像依次通过 Triplet 网络的通道，提取该网络最后一个全连接层的输出作为图像的特征表示，则每个图像被映射为一个低维的数值向量.

## 5 实 验

为了评价所提出方法的有效性，我们在若干常用的数据集上进行了实验. 首先，对所提出的图像三元组挖掘方法进行了实验分析，包括图像三元组的准确率、图像生成采用的图像变换参数的敏感性、三元组挖掘计算效率；然后，训练 Triplet 卷积神经网络来获取图像的特征表示；最后，为了验证学得的视觉表示的有效性，我们进行了分类和聚类实验.

### 5.1 实验设置

#### 5.1.1 实验环境与数据集

实验的硬件环境是：Intel(R) Xeon(R) CPU E5645@2.40 GHz，16 GB 内存，NVIDIA's GeForce GTX 1080 并行计算加速器. 使用 Ubuntu 16.04 操作系统，算法采用 Python 语言实现，神经网络算法基于 TensorFlow[30] 实现.

我们分别在以下数据集上进行了实验：两个手写数字图像数据集(MNIST[31] 和 USPS)，两个多视角物体的图像数据集(COIL-20 和 COIL-100[32])，两个人脸数据集(UMIST[33] 和 YTF)，自然图像数据集(CIFAR-10[34])，场景图像数据集(SCENE-15)，这些数据集都提供了真实的类别标注信息. 需要说明的是，本文方法并未使用这些类别标注信息，只用于实验评价. 表 1 中给出了实验中采用的所有数据集的信息，包括样本数量、图像大小、类别个数均在表 1 中列出. 另外，对数据集进行了以下预处理：(1) 由于部分数据集中的图像尺寸不一致，我们将 UMIST 图像集中图像的尺寸统一到 128×128，将 SCENE-15 图像集中图像的尺寸统一到 256×256；(2) 对于 YTF 数据集，与文献[28]中处理方式相同，首先按人名排序，选择前 41 类的人脸数据，人脸对齐之后将图像尺寸统一到 55×55；(3) 将所有图像数据集进行数据归一化，将图像像素值规范化到均值为 0，方差为 1，以有利于神经网络的训练.

**表 1 实验中使用的数据集**

| 数据集 | 图像大小 | 样本数量 | 类别数目 |
|---|---|---|---|
| MNIST | 28×28 | 60000 | 10 |
| CIFAR-10 | 32×32 | 60000 | 10 |
| COIL-20 | 128×128 | 1440 | 20 |
| USPS | 16×16 | 11000 | 10 |
| COIL-100 | 128×128 | 7200 | 100 |
| UMIST | 128×128 | 1012 | 20 |
| YTF | 55×55 | 10000 | 41 |
| SCENE-15 | 256×256 | 4485 | 15 |

#### 5.1.2 图像生成参数设置

如第 3.1 节所述，为了得到图像集 $\boldsymbol{T}_p$，我们使用了一系列的图像变换. 在表 2 中列出了在各个数据集上使用的图像变换操作及取值范围. 对于 CIFAR-10 数据集，其图像类内差异比较大，背景比较复杂，则适当增大图像变换操作的取值范围. 相反，MNIST 数据集类内图像的差异相对较小，背景单一，则要适度减小图像变换操作的取值范围. 以 CIFAR-10 为例，图 5 中展示了部分图像变换结果.

**表 2 图像变换操作参数取值范围(其中单个数值 $x$ 表示取值范围$[-x,x]$，T/F 表示是/否采用)**

| 变换方式 | 旋转 | 水平平移 | 垂直平移 | 缩放 | 剪切 | 水平翻转 | 高斯模糊 | 色调 |
|---|---|---|---|---|---|---|---|---|
| MNIST | 10 | F | F | 0.1 | 0.1 | F | F | F |
| CIFAR-10 | 40 | 0.3 | 0.3 | 0.1 | 0.1 | T | T | 0.1 |
| COIL-20 | 5 | 0.1 | 0.1 | F | F | T | F | F |
| USPS | 10 | F | F | 0.1 | 0.1 | F | F | F |
| COIL-100 | 5 | 0.1 | 0.1 | 0.1 | 0.1 | T | F | F |
| UMIST | 10 | 0.1 | 0.1 | 0.1 | 0.1 | F | F | F |
| YTF | 5 | 0.1 | 0.1 | 0.05 | 0.05 | T | F | F |
| SCENE-15 | 15 | 0.1 | 0.1 | 0.1 | 0.1 | T | F | F |

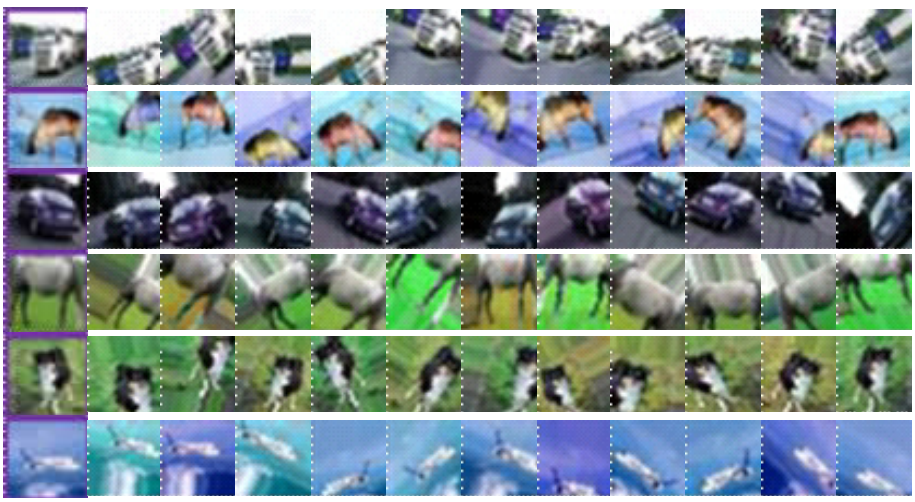


图 5 Cifar-10 数据集上的图像变换生成的图像(其中每一行最左侧的图像是原数据集中的图像)

5.1.3　三元组挖掘参数设置

如第 3 节所述，我们构建二分类卷积网络来挖掘图像三元组样本. 表 3 列出了针对不同数据集设计的二分类卷积网络结构. 接下来我们以 COIL-100 数据集为例，说明该表中各项数据的含义，其它数据集与之类似.

**表 3　三元组挖掘方法中针对不同数据集设计的二分类卷积网络结构**

| 数据集 | 卷积层 | 全连接层 | 卷积核大小 | 卷积特征图个数 | 全连接层神经元 |
|---|---|---|---|---|---|
| MNIST | 3 | 2 | 3-3-3 | 1-32-64-64 | 64-2 |
| CIFAR-10 | 4 | 2 | 3-3-3-3 | 3-64-64-64-64 | 64-2 |
| COIL-20 | 5 | 2 | 3-3-3-3-3 | 1-64-64-64-64-64 | 64-2 |
| USPS | 2 | 1 | 3-3 | 1-32-64 | 2 |
| UMIST | 5 | 2 | 3-3-3-3-3 | 1-64-64-64-64-64 | 64-2 |
| COIL-100 | 5 | 2 | 3-3-3-3-3 | 3-64-64-64-64-64 | 64-2 |
| YTF | 4 | 2 | 5-3-3-3 | 3-64-64-64-64 | 64-2 |
| SCENE-15 | 5 | 2 | 7-5-5-3-3 | 1-64-64-64-64-64 | 64-2 |

如表 3 所示，用于 COIL-100 数据集的二分类卷积网络共有 7 层，前 5 层为卷积层. 卷积核大小“3-3-3-3-3”表示共有 5 个卷积层，每个卷积层的卷积核大小均为 $3\times3$. 卷积特征图个数为“3-64-64-64-64-64”，其中“3”表示 COIL-100 数据集中图像的通道数，其余表示各卷积层的卷积输出特征图的个数. 卷积层的卷积核步长均为 1，对卷积层的输入要进行 0 填充($padding=1$)，保证卷积运算得到的特征图尺寸与卷积输入尺寸相同. 除最后一个卷积层以外的每个卷积层之后，都加上 max-pooling 层，max-pooling 的核大小为 $2\times2$，步长为 2. 网络的后两层为全连接层，全连接第一层有 64 个神经元，第二层即输出层有 2 个神经元. 对于 USPS 数据集，其二分类卷积网络的最后一个卷积层之后添加 max-pooling 层. 由表 3 可知，本文设计的二分类神经网络的结构较为简单，减少了整个神经网络的参数数量，减小了计算负担，使得神经网络更易于被训练，同时也减少了过拟合的可能.

二分类卷积神经网络采用随机梯度下降(SGD)方法进行训练，其超参数包括：学习效率 $\eta$，训练代数 epochs，每个 mini-batch 的大小 batch_size，$L_2$ 正则化参数 $\lambda$，Dropout 参数 $kp$. 各数据集对应的二分类网络使用的超参数保持一致，在表 4 中列出.

**表 4　二分类神经网络训练超参数的选择**

| 参数名 | 参数值 |
|---|---|
| $\eta$ | 0.01 |
| epochs | 3 |
| batch_size | 32 |
| $\lambda$ | 0.001 |
| $kp$ | 0.5 |

5.1.4　Triplet 网络参数设置

如第 4 节所述，我们利用挖掘到的三元组样本训练 Triplet 神经网络. 我们分别在表 1 所示的数据集上进行了实验，表 5 中展示了针对不同数据集设计的 Triplet 卷积网络的结构. 其中，MNIST 和 USPS 数据集中的图像较为简单，我们尽量减少网络的复杂性，而其它数据集对应 Triplet 网络则更深. Triplet 网络的训练同样采用随机梯度下降(SGD)学习方法. 初始学习效率在 0.01 到 0.1 之间调整，并采用学习效率递减策略. 动量下降的参数为 0.9. $L_2$ 正则化参数在 0.001 到 0.005 之间进行调整，Dropout 参数统一为 0.5. Triplet 神经网络训练 15 到 20 epochs，直至损失函数值趋于稳定.

**表 5　针对不同数据集搭建的 Triplet 卷积神经网络结构描述**

| 数据集 | 卷积层 | 全连接层 | 卷积核大小 | 卷积特征图个数 | 全连接层神经元 |
|---|---|---|---|---|---|
| MNIST | 3 | 1 | 3-3-3 | 1-32-64-128 | 128 |
| CIFAR-10 | 4 | 1 | 3-3-3-2 | 3-64-128-256-320 | 256 |
| COIL-20 | 5 | 1 | 3-3-3-3-3 | 1-64-128-192-256-320 | 256 |
| USPS | 2 | 1 | 3-3 | 1-64-128 | 128 |
| UMIST | 5 | 1 | 7-5-3-3-3 | 1-64-128-192-256-320 | 256 |
| COIL-100 | 5 | 1 | 3-3-3-3-3 | 3-64-128-192-256-320 | 128 |
| YTF | 3 | 1 | 3-3-3 | 3-64-128-256 | 128 |
| SCENE-15 | 5 | 1 | 7-5-5-3-3 | 3-63-128-192-256-320 | 128 |

## 5.2　图像三元组挖掘实验结果

5.2.1　三元组质量评估

我们将本文提出的三元组挖掘方法与直接匹配方法和基于 HOG 特征匹配方法进行了比较.

(1) 直接匹配. 以图像数据作为输入，随机选择种子图像之后，选择与种子图像的欧式距离最

小的图像作为正样本，欧式距离最大的图像作为负样本.

(2) 基于 HOG 特征匹配. 首先提取图像的 HOG 特征，然后计算与种子图像 HOG 特征之间的欧式距离最小的图像作为正样本，特征之间欧式距离最大的图像作为负样本.

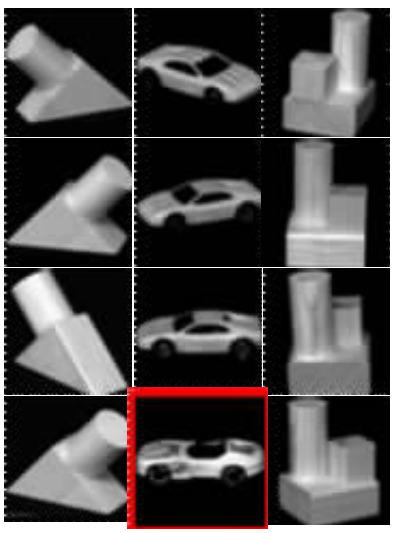
我们使用图像数据集上的真实语义标签作为基准来衡量图像三元组的准确率. 以三元组 $(p, p^+, p^-)$ 为例，若图像 $p$ 与 $p^+$ 属于相同类别，则为真实正样本，使用正样本查准率 $P_p$ 评估；若图像 $p$ 与 $p^-$ 属于不同类别，则为真实负样本，使用负样本查准率 $P_n$ 评估. 为了进一步评估三元组的质量，我们使用三种挖掘方法得到的三元组集合在相同条件下训练 Triplet 神经网络，基于学得的特征表示，使用 $K$-means 聚类算法进行聚类，对聚类结果进行分析.

表 6 中列出了对比实验结果. 由该表可知，尽管直接匹配和基于 HOG 特征的匹配在部分数据集上可以得到更高的查准率，但最终的聚类结果却不如本文提出的方法. 这说明直接匹配和基于 HOG 特征的匹配的方法挖掘的样本较简单，而本文提出的三元组挖掘方法更能抓住语义不变性特征，所挖掘的三元组样本更难，而在表示学习过程中，基于难三元组样本的学习可得到更好的特征表示. 因此，本文所提出的图像三元组挖掘方法可以较为准确地从无标签图像数据集中挖掘出图像三元组信息，从而为表示学习提供更有效的监督信号.

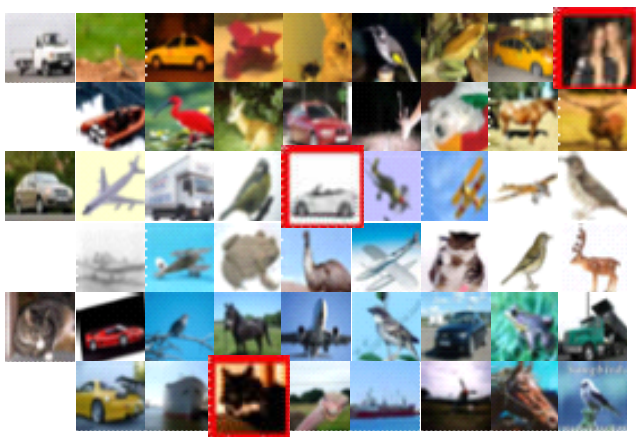
**表 6 三种三元组挖掘方法比较**

| 数据集 | 总数/k | 正样本查准率 $P_p$ | | | 负样本查准率 $P_n$ | | | *NMI* | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | 直接匹配 | 基于 HOG | Proposed | 直接匹配 | 基于 HOG | Proposed | 直接匹配 | 基于 HOG | Proposed |
| COIL-20 | 55 | 0.917 | **0.960** | 0.912 | 0.984 | 0.990 | **0.990** | 0.764 | 0.504 | **0.778** |
| MNIST | 460 | **0.910** | 0.875 | 0.905 | 0.989 | 0.907 | **0.991** | 0.361 | 0.160 | **0.790** |
| CIFAR-10 | 800 | 0.247 | 0.262 | **0.308** | 0.916 | **0.926** | 0.924 | 0.029 | 0.038 | **0.191** |
| USPS | 460 | **0.940** | 0.848 | 0.910 | **0.993** | 0.910 | 0.990 | 0.402 | 0.056 | **0.679** |
| UMIST | 35 | **0.971** | 0.960 | 0.908 | 0.976 | 0.986 | **0.992** | 0.760 | 0.651 | **0.762** |
| COIL-100 | 400 | **0.882** | 0.785 | 0.715 | **0.998** | 0.992 | 0.991 | 0.723 | 0.485 | **0.724** |
| YTF | 460 | **0.997** | 0.989 | 0.852 | 0.970 | **0.993** | 0.957 | 0.695 | 0.658 | **0.801** |
| SCENE-15 | 460 | 0.298 | 0.361 | **0.547** | 0.921 | **0.955** | 0.922 | 0.083 | 0.120 | **0.431** |

实验表明本文提出的无监督图像三元组挖掘方法能传递出较准确的语义信息. 图 6 展示了正样本和负样本示例. 图 6(a)是从 COIL-20 数据集中挖掘出的三元组的正样本示例，每列的第一个图像是查询图像，列中其余图像是与第一个图像相似的图像. 图 6(b)是从 CIFAR-10 数据集中挖掘出的三元组的负样本示例，每行的第一个图像是查询图像，行中其余图像是与第一个图像不相似的图像. 观察发现，图 6(a)中边框所标记的图像与查询图像不属于同一类别，但视觉上却是相似的；同样地，图 6(b)第一行中边框所标记的图像与查询图像“trunk”在视觉上是不相似的，但两者类别标签却相同.



(a) COIL-20数据集的正样本示例



(b) CIFAR-10数据集的负样本示例

图 6 挖掘出的三元组示例

5.2.2　图像变换参数敏感性分析

下面我们分析 5.1.2 节描述的各个图像变换操作参数的取值范围对挖掘得到的三元组准确率的影响. 如图 7 所示，在 MNIST 数据集上，挖掘得到的三元组的真实负样本查准率受各图像变换操作的取值范围影响较小，而真实正样本查准率受图像变换操作的取值范围影响较大. 原因分析如下：正样本在图像集中分布更为稀疏，正确率相比负样本要低. 图像变换操作的对象是种子图像，因此各个操作的参数范围的变化对学习负样本所在类别的不变性特征基本无影响. 观察图 7 中真实正样本曲线，恰好反映了数据集类内图像在目标的角度、位置、大小等方面的分布情况. 如图 7(a)中，旋转角度在 10 度时真实正样本查准率达到最高，则说明 MNIST 数据集中类内图像的角度大约在 10 度左右. 在其它数据集上也进行了敏感性实验，在真实正样本查准率达到最

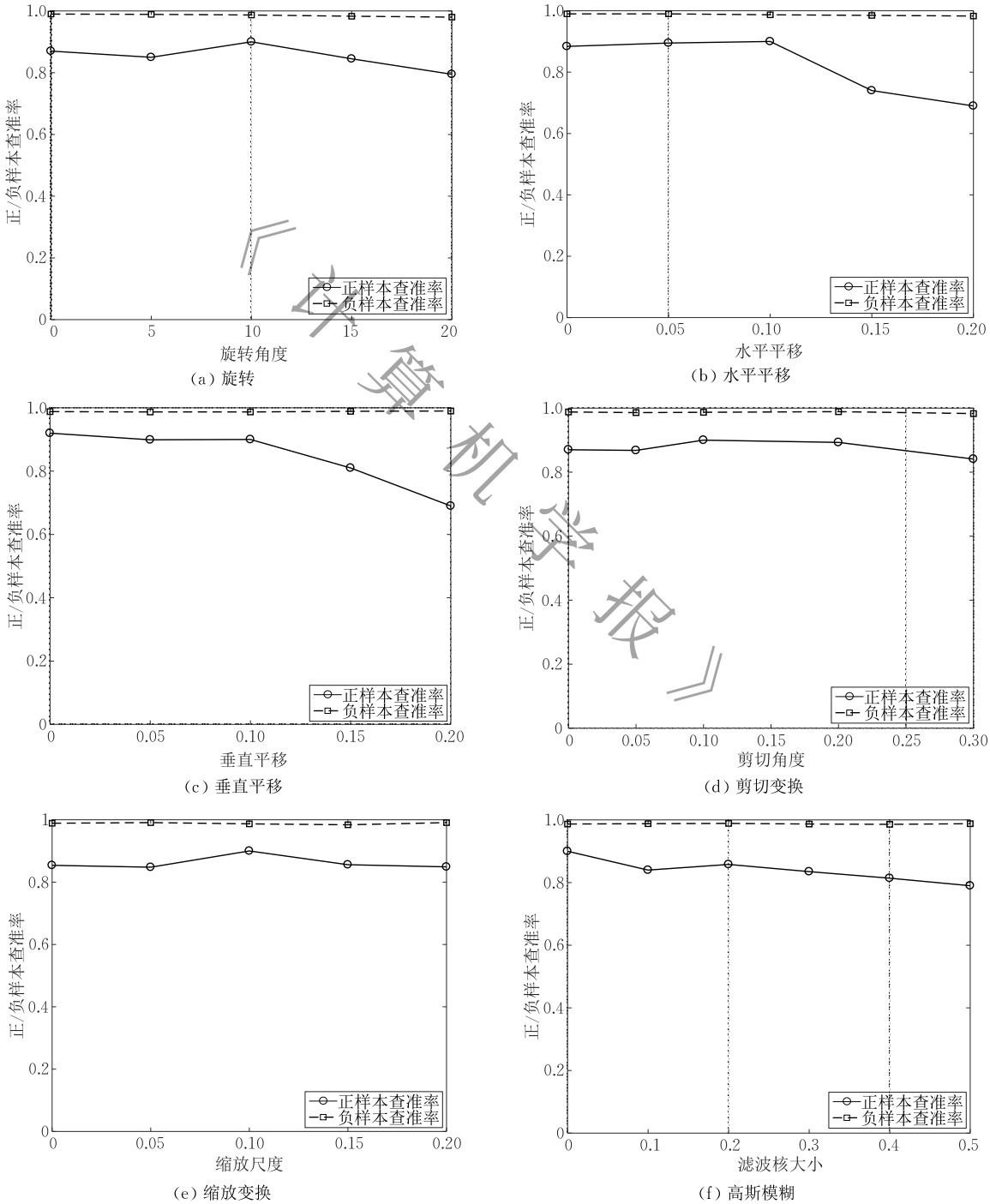
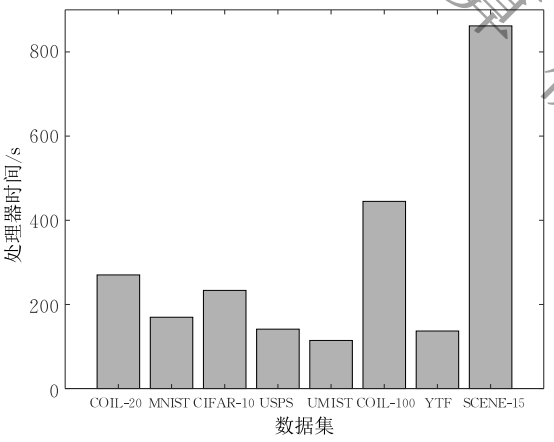



图 7　MNIST 数据集上挖掘三元组查准率与各图像变换参数范围的关系(实线和虚线分别表示正样本查准率和负样本查准率)

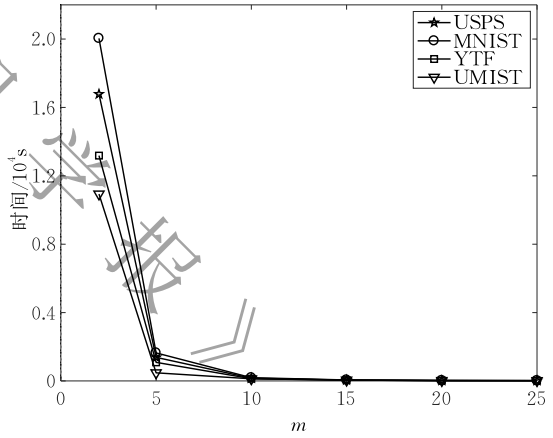
高时，各图像变换参数的取值与表 2 中所列值相近，真实负样本查准率受图像变换参数影响较小. 上述敏感性实验结果说明，本文提出的图像三元组挖掘方法在不同图像数据集下是鲁棒的.

5.2.3 三元组挖掘时间

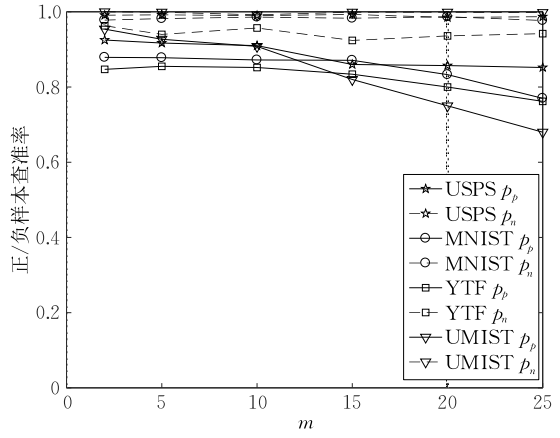
下面我们分析各实验数据集下的三元组挖掘算法的时间消耗. 据算法 1，基于一个种子图像，训练二分类网络，进行 $m$ 次随机采样并预测可得到 $m$ 个三元组，进而利用直递性快速扩充图像三元组的数量. 图 8 展示了三元组挖掘算法的计算效率分析结果，我们通过实验分析了算法中参数 $m$ 对三元组挖掘速度和准确率的影响. 观察图 8(b)，当参数 $m$ 逐渐增大时，挖掘 10k 个图像三元组所消耗的时间呈指数下降，如参数 $m$ 自 2 增大到 10 时，所消耗的时间从小时量级迅速下降到秒级. 另外，从图 8(c)可以看出，随着参数 $m$ 逐渐增大时，所挖掘的三元组的正样本查准率呈缓慢下降趋势，对应负样本查准率则在小范围内保持相对稳定. 由此可知，变量 $m$ 对三元组正样本查准率和时间消耗的关系：若偏向于更快的挖掘速度，则可适当地增大参数 $m$，若偏向于获得更高的三元组正样本准确率，则应减小参数 $m$. 因此算法在应用中，可根据实际情况调整该参数进行权衡. 图 8(a)展示了在相同条件下各数据集上挖掘一定数量的三元组样本所消耗的时间. 实验中设定 $m=10$，从各数据集中随机选取 22 个种子图像，基于每个种子图像训练二分类神经网络，预测 10 次并利用直递性快速扩充三元组，得到 10120 个(约 10k)图像三元组. 实验结果显示，在 8 个数据集下，挖掘 10k 个三元组样本所消耗的时间均在百秒级别. 通过对比各数据集的时间消耗，影响三元组挖掘算法性能的主要因素是图像的尺寸和通道数，大尺寸图像和彩色图像会加大三元组挖掘模型的运算量，进而增大了所需的时间. 进一步分析，通常情况下，在大数据集下需要更多的三元组样本以学习其更好的特征表示，我们对三元组样本的数量与时间的关系进行了分析. 图 8(d)展示了在 USPS、MNIST、YTF、UMIST 数据集中挖掘不同数量的三元组样本与所消耗时间的关系，可以看出消耗的时间与三



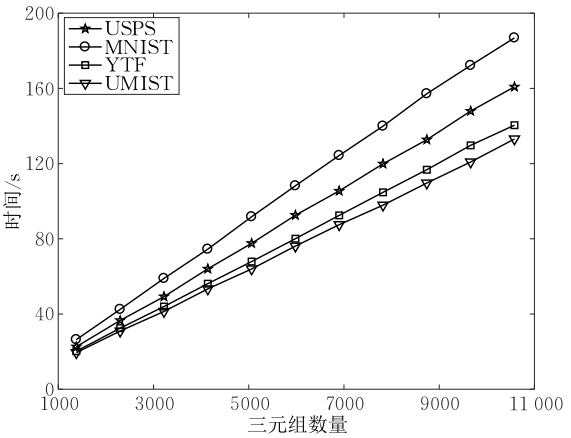

(a) 各数据集10k三元组的挖掘时间对比




(b) 不同参数$m$下挖掘10k三元组的时间消耗




(c) 不同$m$值对三元组准确率的影响




(d) 挖掘不同三元组数量所消耗的时间，$m=10$

图 8 三元组挖掘算法时间消耗分析

元组样本数量呈线性增长关系.由此可以得出,本文提出的三元组挖掘方法可以适用于大数据集.

### 5.3　聚类实验

使用训练好的 Triplet 神经网络提取整个数据集的特征表示.为了验证所学得的特征表示的有效性,我们首先将特征表示应用于聚类问题,以其在聚类上的表现对表示的有效性进行衡量.聚类结果使用 $NMI$[35] 和 $ARI$[36] 准则进行评价.

$NMI$(Normalized Mutual Information):

$$NMI(U,V)=\frac{MI(U,V)}{\sqrt{H(U)H(V)}} \tag{11}$$

其中,$H(U)$表示熵,$MI(U,V)$表示互信息.$NMI$ 取值范围为[0,1],$NMI=1$ 表示聚类结果达到最佳,$NMI=0$ 则表示聚类结果最差.

$ARI$(Adjusted Rand index):

$$ARI=\frac{RI-E[RI]}{\max(RI)-E[RI]} \tag{12}$$

若两个数据点属于同一聚类簇且具有相同的类别标签,或分属于不同聚类簇且具有不同的类别标签,则称这两个数据点在聚类结果与真实类别标注达成一致.式(12)中,$RI$ 表示所有成对数据点达成一致所占的比例.$ARI$ 的取值范围是[−1,1],$ARI$ 取负值或小的正值表示聚类结果较差.当随机划分聚类簇时,$ARI$ 的取值趋近于 0,$ARI=1$ 表示聚类效果达到最佳.

#### 5.3.1　不同聚类算法实验结果

我们选择三种基准聚类算法,包括 $K$-means 聚类算法(KM)、标准谱聚类算法[37](SC),凝聚聚类算法(AC-link).我们分别对比了基准聚类算法以原始图像作为输入和以学得的表示作为输入时的聚类表现.每次实验分别运行 5 次,取平均值,实验结果在表 7 中列出.可以看到,以学得的表示作为基准聚类算法的输入时,聚类结果都优于以原始图像作为输入的结果,$K$-means 聚类算法的 $NMI$ 和 $ARI$ 平均提升 14.89%和 15.3%,标准谱聚类算法 $NMI$ 和 $ARI$ 平均提升 14.3%和 14.98%,凝聚聚类算法 $NMI$ 和 $ARI$ 平均提升了 11.86%和 11.92%.由此可以得出结论:本文方法学得的视觉表示具有良好的判别性,对基准聚类算法具有较大提升效果.

**表 7　学习得到的特征表示与原始图像作为输入时在不同基准聚类算法上的比较**

| 数据集 | Representation+KM | | KM | | Representation+SC | | SC | | Representation+AC | | AC | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *NMI* | *ARI* | *NMI* | *ARI* | *NMI* | *ARI* | *NMI* | *ARI* | *NMI* | *ARI* | *NMI* | *ARI* |
| CIFAR-10 | **0.191** | **0.177** | 0.064 | 0.030 | 0.122 | 0.119 | 0.067 | 0.033 | 0.113 | 0.102 | 0.061 | 0.045 |
| MNIST | **0.790** | **0.768** | 0.485 | 0.402 | 0.767 | 0.756 | 0.475 | 0.451 | 0.783 | 0.753 | 0.662 | 0.611 |
| COIL-20 | **0.778** | **0.767** | 0.766 | 0.747 | 0.727 | 0.697 | 0.622 | 0.617 | 0.710 | 0.672 | 0.512 | 0.488 |
| COIL-100 | 0.724 | 0.645 | 0.706 | 0.639 | **0.727** | **0.664** | 0.592 | 0.573 | 0.726 | 0.646 | 0.711 | 0.652 |
| USPS | **0.679** | **0.672** | 0.485 | 0.470 | 0.640 | 0.635 | 0.575 | 0.560 | 0.632 | 0.623 | 0.579 | 0.568 |
| UMIST | 0.762 | 0.720 | 0.609 | 0.579 | 0.748 | 0.718 | 0.611 | 0.601 | **0.764** | **0.724** | 0.643 | 0.586 |
| YTF | 0.801 | 0.785 | 0.744 | 0.733 | 0.746 | 0.709 | 0.703 | 0.609 | **0.834** | **0.825** | 0.798 | 0.788 |
| SCENE-15 | 0.431 | 0.388 | 0.106 | 0.098 | 0.410 | 0.379 | 0.096 | 0.034 | **0.452** | **0.437** | 0.099 | 0.090 |

#### 5.3.2　与人工设计特征对比实验

为了进一步检验本文提出方法所学得的特征表示的判别性,我们将其与 GIST、HOG 特征表示进行对比.实验中,我们选择 $K$-means 算法作为基准聚类算法,分别以 GIST、HOG 特征以及本文所学得的特征表示作为聚类算法的输入,以聚类的结果对各个特征表示的判别能力进行分析.表 8 中列出了实验结果.通过对比可以发现,GIST 和 HOG 作为优秀的传统视觉特征描述算子,对 $K$-means 聚类算法有较好的提升效果,而基于 GIST 特征的聚类结果优于基于 HOG 特征的聚类结果,其聚类表现 $NMI$ 和 $ARI$ 评价值分别高于 HOG 特征 8.05%和 10.5%;进一步,本文所提出方法学得的特征表示的聚类结果要优于 GIST 和 HOG 特征表示方法,比 GIST 特征聚类结果 $NMI$ 和 $ARI$ 值平均提升分别为 8.7%和 7.6%,比 HOG 特征聚类结果 $NMI$ 和 $ARI$ 值平均提升分别为 16.7%和 18.1%;另一方面,本文所学得的特征表示比有效的 GIST 特征和 HOG 特征具有更低的维度.综上可得,本文方法学得的特征表示比 HOG 和 GIST 特征更具判别性,且更低维度的特征有利于降低后续其它视觉算法的计算复杂度.

表 8 学习得到的特征表示与 GIST、HOG 特征在 *K*-means 聚类上的对比

| 数据集 | GIST | | | HOG | | | Proposed | | |
|---|---|---|---|---|---|---|---|---|---|
| | 特征维度 | *NMI* | *ARI* | 特征维度 | *NMI* | *ARI* | 特征维度 | *NMI* | *ARI* |
| CIFAR-10 | 960 | 0.167 | 0.159 | 324 | 0.097 | 0.074 | 256 | **0.191** | **0.177** |
| MNIST | 960 | 0.559 | 0.556 | 324 | 0.263 | 0.149 | 128 | **0.790** | **0.768** |
| COIL-20 | 960 | 0.742 | 0.717 | 324 | 0.761 | 0.712 | 256 | **0.778** | **0.767** |
| COIL-100 | 960 | **0.813** | **0.757** | 324 | 0.737 | 0.669 | 128 | 0.724 | 0.645 |
| USPS | 960 | 0.534 | 0.527 | 324 | 0.464 | 0.453 | 128 | **0.679** | **0.672** |
| UMIST | 960 | 0.584 | 0.547 | 324 | 0.564 | 0.527 | 256 | **0.762** | **0.720** |
| YTF | 960 | 0.733 | 0.726 | 576 | 0.763 | 0.753 | 128 | **0.801** | **0.785** |
| SCENE-15 | 960 | 0.326 | 0.320 | 324 | 0.165 | 0.131 | 256 | **0.431** | **0.388** |

5.3.3 与其它无监督视觉表示空间对比

接下来，将所提出的方法与其它的无监督方法进行对比. 我们分析对比了四个表示空间，包括原始图像空间、主成分分析（PCA）空间、Auto-Encoder 学习得到的非线性映射空间和本文所提出的无监督方法学习得到的表示空间（以下分别简称原始空间、PCA 空间、Auto-Encoder 空间、本文空间）. 分别在这四个空间中使用 *K*-means 算法进行聚类，在表 9 中列出了 MNIST 数据集上的实验结果.

表 9 MNIST 数据集上 4 个不同表示空间中的 *K*-means 聚类结果

| 表示空间 | *NMI* | *ARI* |
|---|---|---|
| Original | 0.485 | 0.475 |
| PCA | 0.285 | 0.270 |
| Auto-Encoder | 0.660 | 0.630 |
| Proposed | **0.790** | **0.768** |

通过对比看出，PCA 空间比原始空间中的聚类结果要差；Auto-Encoder 空间中的聚类表现要优于原始空间；而本文空间中的聚类表现优于 Auto-Encoder 空间. 原因分析如下：*K*-means 算法是典型的基于距离的聚类算法，采用距离衡量对象相似性，对象距离越近，其相似度越大，算法将距离靠近的对象聚为一个聚类簇. Auto-Encoder 模型的编码层和解码层要求实现编码和解码的过程，并未试图使相似的输入数据在编码层具有相近的编码，不相似输入数据具有相异的编码. 而本文所提出的方法在学习特征表示时以提升判别能力为目标，因此其聚类表现优于 Auto-Encoder 表示空间.

## 5.4 分类实验

下面进一步通过分类实验验证学得的表示具有良好的判别性. 我们构建了单层神经网络和线性 SVM 分类器对多个数据集进行分类. 以学得的特征表示作为两个分类器的输入，采用 10-交叉验证的方法测试分类准确率. 值得一提的是，在训练分类器的时候，我们只简单地选择合理的学习参数，没有进行精细的参数调优和使用其它优化策略，尽管如此，依然取得了具有竞争力的实验结果. 我们与各个数据集上已知的最佳分类结果进行对比，表 10 中列出了对比结果. 在监督学习最优结果一列中，带星号“*”的数据是由我们自己实现的结构类似 AlexNet 的卷积神经网络的最佳分类结果. 从实验可以看出，本文方法结合线性 SVM 分类器，在 UMIST 和 COIL-100 数据集上已经达到据我们所知目前最好的分类结果，在 USPS 和 YTF 数据集上接近于目前最好的监督学习方法的分类结果，在其它数据集上本文结果稍弱于目前最好的监督学习方法的分类结果. 原因分析如下：本文的学习目标是通过无监督方式学习具有判别性的特征表示，并不是以分类任务的准确率作为直接的优化目标. 实验结果显示，基于本文方法所学得的特征表示，使用基本的线性分类器也能取得有竞争力的分类结果. 因此得出以下结论：本文所提出的方法学得的特征表示具有良好的判别性.

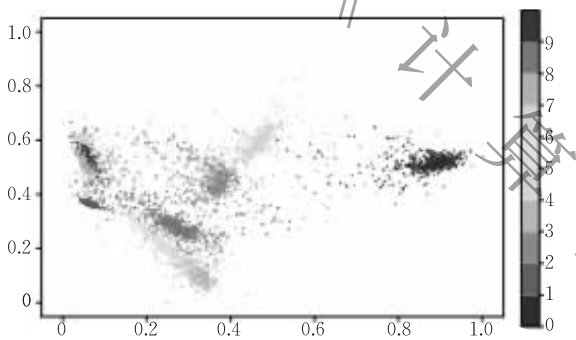
表 10 基于学习得到的特征表示的分类实验结果

| 数据集 | 表示＋1 layer NN | 表示＋linear SVM | 监督学习最优结果 |
|---|---|---|---|
| COIL-20 | 0.843 | 0.923 | **0.997**[38] |
| MNIST | 0.907 | 0.893 | **0.997**[39] |
| USPS | 0.854 | 0.976 | **0.980**[39] |
| UMIST | 0.860 | **0.990** | 0.980[40] |
| COIL-100 | 0.726 | **0.984** | 0.963[41] |
| CIFAR-10 | 0.764 | 0.736 | **0.965**[42] |
| YTF | 0.923 | 0.907 | **0.967*** |
| SCENE-15 | 0.856 | 0.844 | **0.945**[43] |

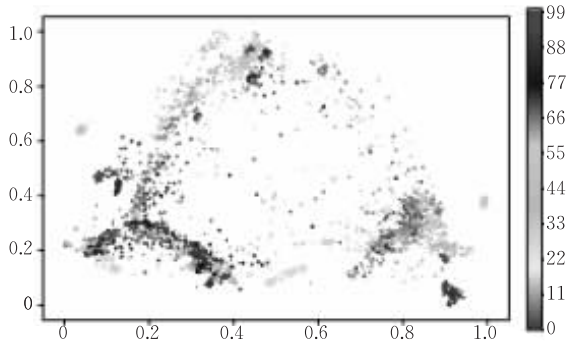
## 5.5 视觉表示可视化

为了验证本文所提出的方法能将图像嵌入到有意义的欧式空间中，我们利用 PCA 将所习得的图像表示投影到二维欧式空间中并进行可视化. 我们分别在 MNIST（图 9(a)）、COIL-100（图 9(b)）、USPS（图 9(c)）数据集上进行了表示可视化实验，PCA 取前两个主向量为投影方向. 从图 9 中我们可以看到，

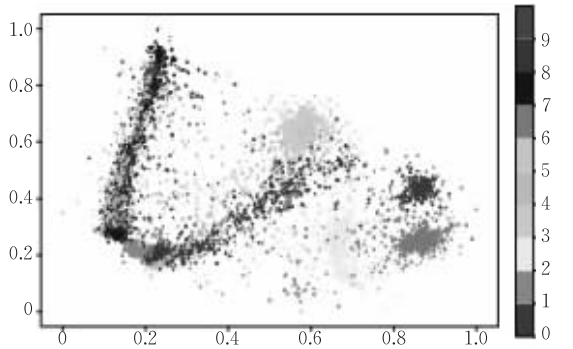
本文所提出的方法学得的特征表示具有良好的判别性. 总体上，表示空间中同一类别的图像分布比较集中，而不同类别的图像则相互远离. 需要指出的是，某些类内图像在表示空间中分布仍不够紧凑，甚至呈带状分布. 原因分析如下：(1) 在实验的过程中，相比于图像三元组总量，我们挖掘的用于表示学习的三元组只占很少的一部分；(2) PCA 投影丢失了一定的信息，导致数据在二维投影面产生了一定的重叠. 本文所提出的方法通过挖掘图像三元组，并以学习的方式将图像非线性嵌入到低维表示空间. 相较于图像原始空间、传统特征空间或无监督表示空间，本文表示空间扩大类间差异性，缩小类内差异性，使数据集在本文表示空间中更具判别性. 因此，本文提出的方法学到的特征表示在应用到聚类和分类问题时，可以起到较大的性能提升作用.



(a) MNIST-test



(b) COIL-100



(c) USPS

图 9　特征表示的 PCA 可视化

## 6　结　论

本文提出了一种基于图像三元组挖掘的无监督视觉表示学习方法，用于从无标记图像数据集中自动获得图像的特征表示. 该方法基于二分类神经网络从无标记图像集合中挖掘出图像三元组，并利用图像三元组训练 Triplet 卷积神经网络以学习图像的特征表示. 此方法框架是完全无监督的，没有使用任何图像标记信息. 在多个图像数据集上的聚类任务中，学得的表示对于基准聚类算法的提升效果最高达 15.3%；相比于传统视觉特征方法，学得的表示也取得了约 12.7%的性能提升，相比于基本 Auto-Encoder 表示空间，本文表示空间具有更好的判别性. 在分类问题上，基于学得的视觉表示，采用基本线性分类器，在 UMIST、COIL-100 数据集上取得了据我们所知的目前最好的分类结果，在 MNIST、COIL-20、USPS、YTF 数据集上也取得了有竞争力的结果. 这些结果表明，我们所提出的方法可以较准确地从图像数据集中挖掘出图像三元组，进一步利用三元组蕴含的相似性信息所学得的特征表示具有良好的判别性，可以有效地应用到图像聚类、分类和检索等任务中.

## 参　考　文　献


[1] Dalal N, Triggs B. Histograms of oriented gradients for human detection//Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Diego, USA, 2005: 886-893

[2] Lowe D G. Object recognition from local scale-invariant features//Proceedings of the IEEE International Conference on Computer Vision. Kerkyra, Greece, 1999: 1150-1157

[3] Ojala T, Harwood I. A comparative study of texture measures with classification based on feature distributions. Pattern Recognition, 1996, 29(1): 51-59

[4] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks//Proceedings of the International Conference on Neural Information Processing Systems. Lake Tahoe, USA, 2012: 1097-1105

[5] Zhou Fei-Yan, Jin Lin-Peng, Dong Jun. Review of convolutional neural network. Chinese Journal of Computers, 2017, 40(6): 1229-1251(in Chinese)
(周飞燕，金林鹏，董军. 卷积神经网络研究综述. 计算机学报，2017，40(6)：1229-1251)

[6] Bengio Y, Courville A, Vincent P. Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2013, 35(8): 1798

[7] Jiao Li-Cheng, Yang Shu-Yuan, Liu Fang, et al. Seventy years beyond neural networks: Retrospect and prospect. Chinese Journal of Computers, 2016, 39(8): 1697-1716(in Chinese)
(焦李成，杨淑媛，刘芳等. 神经网络七十年：回顾与展望. 计算机学报，2016，39(8)：1697-1716)

[8] Hoffer E, Ailon N. Deep metric learning using triplet network //Proceedings of the International Workshop on Similarity-Based Pattern Recognition. Copenhagen, Denmark, 2015: 84-92

[9] Chopra S, Hadsell R, Lecun Y. Learning a similarity metric discriminatively, with application to face verification// Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Diego, USA, 2005: 539-546

[10] Wang J, Song Y, Leung T, et al. Learning fine-grained image similarity with deep ranking//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Columbus, USA, 2014: 1386-1393

[11] Zhang R, Lin L, Zhang R, et al. Bit-scalable deep hashing with regularized similarity learning for image retrieval and person re-identification. IEEE Transactions on Image Processing of the IEEE Signal Processing Society, 2015, 24(12): 4766-4779

[12] Hu J, Lu J, Tan Y P. Discriminative deep metric learning for face verification in the wild//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Columbus, USA, 2014: 1875-1882

[13] Le Q V. Building high-level features using large scale unsupervised learning//Proceedings of the IEEE International Conference on Speech and Signal Processing (ICASSP). Vancouver, Canada, 2013: 8595-8598

[14] Marc' Aurelio Ranzato, Huang F J, Boureau Y, et al. Unsupervised learning of invariant feature hierarchies with applications to object recognition//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Minneapolis, USA, 2007: 1-8

[15] Schölkopf B, Platt J, Hofmann T. Greedy layer-wise training of deep networks//Proceedings of the International Conference on Neural Information Processing Systems. Hong Kong, China, 2006: 153-160

[16] Lin K, Lu J, Chen C S, et al. Learning compact binary descriptors with unsupervised deep neural networks// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 1183-1192

[17] Dosovitskiy A, Fischer P, Springenberg J T, et al. Discriminative unsupervised feature learning with exemplar convolutional neural networks. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2014, 38(9): 1734

[18] Xie J, Girshick R, Farhadi A. Unsupervised deep embedding for clustering analysis//Proceedings of the International Conference on Machine Learning. New York, USA, 2016: 478-487

[19] Ng A. Sparse autoencoder. CS294A Lecture Notes. Stanford University, USA, 2011

[20] Vincent P, Larochelle H, Bengio Y, et al. Extracting and composing robust features with denoising autoencoders// Proceedings of the International Conference on Machine Learning. Helsinki, Finland, 2008: 1096-1103

[21] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks. Science, 2006, 313(5786): 504-507

[22] Masci J, Meier U, Dan C, et al. Stacked convolutional auto-encoders for hierarchical feature extraction//Proceedings of the Artificial Neural Networks and Machine Learning. Espoo, Finland, 2011: 52-59

[23] Wang X, Gupta A. Unsupervised learning of visual representations using videos//Proceedings of the IEEE International Conference on Computer Vision. Santiago, Chile, 2015: 2794-2802

[24] Doersch C, Gupta A, Efros A A. Unsupervised visual representation learning by context prediction//Proceedings of the IEEE International Conference on Computer Vision. Santiago, Chile, 2015: 1422-1430

[25] Li D, Hung W C, Huang J B, et al. Unsupervised visual representation learning by graph-based consistent constraints //Proceedings of the European Conference on Computer Vision. Amsterdam, The Netherlands, 2016: 678-694

[26] Misra I, Zitnick C L, Hebert M. Shuffle and learn: Unsupervised learning using temporal order verification//Proceedings of the European Conference on Computer Vision. Amsterdam, The Netherlands, 2016: 527-544

[27] Jayaraman D, Grauman K. Learning image representations tied to ego-motion//Proceedings of the IEEE International Conference on Computer Vision (ICCV). Santiago, Chile, 2017: 1413-1421

[28] Yang J, Parikh D, Batra D. Joint unsupervised learning of deep representations and image clusters//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 5147-5156

[29] Liang X, Liu S, Wei Y, et al. Towards computational baby learning: A weakly-supervised approach for object detection// Proceedings of the IEEE International Conference on Computer Vision. Santiago, Chile, 2015: 999-1007

[30] Abadi M, Agarwal A, Barham P, et al. TensorFlow: Large-scale machine learning on heterogeneous distributed systems. arXiv:1603.04467, 2016

[31] Lécun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, 86(11): 2278-2324

[32] Nene S A, Nayar S K, Murase H. Columbia object image library (coil-100). Columbia University, New York, USA: Technical Report CUCS-005-96, 1996

[33] Graham D B, Allinson N M. Characterising Virtual Eigensignatures for General Purpose Face Recognition. Berlin, Germany: Springer, 1998

[34] Krizhevsky A, Hinton G. Learning Multiple Layers of Features from Tiny Images [M. S. dissertation]. Department of Computer Science, University of Toronto, Canada, 2009

[35] Li Z, Yang Y, Liu J, et al. Unsupervised feature selection using nonnegative spectral analysis//Proceedings of the Association for the Advancement of Artificial Intelligence. Toronto, Canada, 2012: 1026-1032

[36] Hubert L, Arabie P. Comparing partitions. Journal of Classification, 1985, 2(1): 193-218

[37] Yu S X, Shi J. Multiclass spectral clustering//Proceedings of the IEEE International Conference on Computer Vision. Beijing, China, 2003: 313-319

[38] Sihag S, Dutta P K. Faster method for deep belief network based object classification using DWT. arXiv: 1511.06276, 2015

[39] Wan L, Zeiler M, Zhang S, et al. Regularization of neural networks using DropConnect//Proceedings of the International Conference on Machine Learning. Atlanta, USA, 2013: 1058-1066

[40] Lu C Y, Huang D S. Optimized projections for sparse representation based classification. Neurocomputing, 2013, 113(10): 213-219

[41] Neves-Silva R, Tsihrintzis G A, Uskov V. Smart Digital Futures 2014. Gloucester, UK: IOS Press, 2014

[42] Graham B. Fractional max-pooling. arXiv preprint arXiv: 1412.6071, 2014

[43] Khan S H, Hayat M, Bennamoun M, et al. A discriminative representation of convolutional features for indoor scene recognition. IEEE Transactions on Image Processing, 2016, 25(7): 3372-3383

**HE Guo-Cai**, born in 1993, M. S. candidate. His research interests include machine learning, computer vision.

**LIU Xia-Bi**, born in 1972, Ph. D., associate professor. His research interests include machine learning, pattern recognition, computer vision, and information retrieval.



**Background**

Recently, the representation learning has attracted a lot of attention in the fields of machine learning, computer vision, natural language processing (NLP), and etc. So Bengio and Lecun started the International Conference on Learning Representations (ICLR) in 2013. There are two types of representation learning methods, i. e., the supervised and the unsupervised. To our knowledge, the supervised strategy has been studied more thoroughly than unsupervised one. However, much more data are unlabeled in applications, the unsupervised representation learning should be explored more.

This paper focuses on unsupervised learning of visual representation. We propose a novel unsupervised deep learning method to solve the representation problem of static images. There are two stages involved in our method. Firstly, we use the convolutional neural network (CNN) to mine triplet samples from image datasets in an unsupervised manner. Then, these image triplets are used for training the Triplet CNN to obtain visual representations. Through careful experiments, we show that the proposed unsupervised method for learning visual representation is effective.

This research is supported by the National Natural Science Foundation of China (60973059, 81171407) and the Program for New Century Excellent Talents in University of China (NCET-10-0044).