基于超图排序算法的视频摘要

冀 中,樊帅飞
(天津大学电子信息工程学院,天津 300072)

摘 要: 视频摘要技术作为一种快速感知视频内容的方式得到了广泛的关注.现有基于图模型的视频摘要方法将视频帧作为顶点,通过边表示两个顶点之间的关系,但并不能很好地捕获视频帧之间的复杂关系.为了克服该缺点,本文提出了一种基于超图排序算法的静态视频摘要方法(Hyper-Graph Ranking based Video Summarization,HGRVS).HGRVS方法首先通过构建视频超图模型,将任意多个有内在关联的视频帧使用一条超边连接;然后提出一种基于超图排序的视频帧分类算法将视频帧按内容分类;最后通过求解提出的一种优化函数来生成静态视频摘要.在Open Video Project和YouTube两个数据集上的大量主观与客观实验验证了所提HGRVS算法的优良性能.

关键词: 视频摘要; 超图; 超图排序; 视频帧分类; 关键帧提取
中图分类号: TN919.8 **文献标识码:** A **文章编号:** 0372-2112 (2017)05-1035-09
电子学报URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2017.05.002

Video Summarization with Hyper-Graph Ranking

JI Zhong, FAN Shuai-fei
(*School of Electronic Information Engineering, Tianjin University, Tianjin* 300072, *China*)

Abstract: Video summarization has received widely attention as a technique to quickly display the main video content. Existing graph model based method takes video frames as vertices and uses the edges to build the relationship between two vertices, which may not well capture the complex relationship among the video frames. To overcome this drawback, we present a novel method based on hyper-graph ranking to generate a static video summarization, and name it Hyper-Graph Ranking based Video Summarization (HGRVS). Specifically, HGRVS first builds a video hyper-graph model to connect the video frames that have internal relations with hyper-edges; then classifies the video frames with an idea borrowed from the hyper-graph ranking method to provide the candidate keyframes; the video summarization is finally determined with the keyframes chosen by an objective function. Extensive subjective and objective experiments on the popular Open Video Project and YouTube datasets clearly demonstrate the superiority of HGRVS to the state-of-the-art approaches.

Key words: video summarization; hyper-graph; hyper-graph ranking; video frames classification; keyframe extraction

1 引言

海量视频的存在与不断增长使得如何准确获取并高效呈现视频的主要内容成为当前研究的热点与难点.视频摘要(Video Summarization)技术是解决此问题的一类有效方法.它以一种简略的形式将视频内容表示出来,是对目标视频内容的一种总结.

本文重点研究基于关键帧的静态视频摘要,其方法主要包括基于聚类的方法[1~3],基于图模型的方法[4~10],基于重构的方法[11],基于矩阵分解的方法[12],等.基于聚类的方法是通过聚类的方式将相似的视频帧进行分类然后生成视频摘要.例如,Furini等人[2]提出了一种称为STIMO(STIll and Moving Video Storyboard)的方法,该方法首先提取视频每一帧的颜色特征,然后通过FPF(Farthest Point-First)算法对视频帧进行聚类,最后去除无意义的冗余关键帧.DeAvila等人[3]提出了VSUMM(Video SUMMarization)方法,通过对视频的颜色特征进行k均值聚类,并且选取聚类中心作为关键帧,从而生成视频摘要.基于图模型的方法通过顶点与边建模视频帧之间的关系,通过相关目标函数的优化生成视频摘要.例如,Mundur等人[4]首先对视频采样,提取采样视频帧的颜色特征,然后将每一个采样视

收稿日期:2016-01-26;修回日期:2016-11-28;责任编辑:马兰英
基金项目:国家自然科学基金(No.61472273)

频帧看作德劳内(Delaunay)图的顶点并构建德劳内图,通过移除德劳内图中的部分边进行聚类,最后选取距离聚类中心最近的一帧作为关键帧. 随后,Kuanar 等人[5]改进了[4]中的方法,在移除德劳内图中的部分边进行聚类时,加入了结构约束,进一步保证了能够更好地保留类内的边,同时移除类间的边. Zhai 等人[6]利用视频帧构建 k 近邻图,然后将其分割为多个子图,最后通过一个混合模型选择关键帧. Panda 等人[10]将视频帧作为图的顶点,构建骨架(skeleton)图,通过最小生成树(MST)的方法对顶点进行聚类,然后对聚类结果按照类内视频帧的个数排序,并选取关键帧集合. 此外,近年还出现了基于新视角合成的方法[13],通过为视频内容构建新的观察视角,增加视频浏览空间,从本质上减少视频内容的重复和遮挡,并根据摘要内容实时推荐并调整到最优观察视角.

近年来,图及超图模型在图像检索[14]、文本查询[15]、视觉追踪[16,17]等领域均取得了显著成果. 在图排序学习方面,Deng 等人[18,19]提出了 Co-RMGL(Co-Regularization Multi-Graph Learning)算法用于图像及视觉重排序等方面,Co-RMGL 算法通过提取图像多种特征并将每种特征构建对应的图模型,然后对图内(intra-graph)和图间(inter-graph)分别添加约束并对多图进行融合,通过有监督的学习完成排序. Yang 等人[16]首次将多图排序模型用于视觉追踪领域,提取多种视觉特征构建多图并融合到一个多图排序框架中,通过动作模型对图像区域采样并利用多图排序框架排序,将最高得分区域认为是追踪的物体. 在超图排序学习方面,超图排序的基本思想是通过将相似的顶点用超边连接,然后发现顶点之间的关系,使得与查询顶点在同一条超边内的顶点具有较高的排序得分. 例如,Huang 等人[14]提出了一种基于超图排序的图像检索算法,将图像作为超图的顶点,相似图像利用超边连接,检索图像作为标注顶点,通过超图的直推式学习计算未标注图像相对于标注图像的得分,从而达到检索图像的目的. Wang 等人[15]将文本看做超图顶点,相似主题的文本使用超边连接,将查询文本作为标注顶点,利用超图排序模型提出了一种半监督的文本排序算法. Lu 等人[17]基于超图的排序提出了视觉追踪系统,通过构建三种类型的超图并进行线性的融合,将物体的追踪转换为超图的直推式学习,追踪物体的最优位置通过排序得分来决定. Li 等人[20]提出一种新闻推荐系统,通过建立包含 4 种含义的顶点和 8 中关系超边的超图模型,通过超图排序向用户推荐用户感兴趣的新闻.

现有基于图模型的视频摘要方法图的一条边只能连接两个视频帧,较难捕获视频帧之间更复杂的关系. 为了克服这个不足,本文使用超图(Hyper-graph)建模视频帧之间的关系. 由于超图是对简单图的扩展,其超边可以包含任意多的顶点,因此更适合用于视频摘要. 为此,本文提出了一种新颖的静态视频摘要方法——基于超图排序的视频摘要算法(Hyper-Graph Ranking based Video Summarization,HGRVS),该方法首先借鉴超图排序的思想将视频帧进行分类,目的是获得候选关键帧,然后通过求解一种目标函数最终确定关键帧以形成摘要,算法整体框图如图 1 所示.

本文的贡献主要有以下 3 点:(1)首次将超图模型应用于视频摘要领域. (2)提出一种基于超图排序思想的视频帧分类算法,首先将视频帧进行得分排序,根据得分将视频帧进行分类,以得到候选关键帧. (3)提出一种综合考虑多样性以及信息最大覆盖性等方面的关键帧确定方法,最终确定并呈现视频摘要.

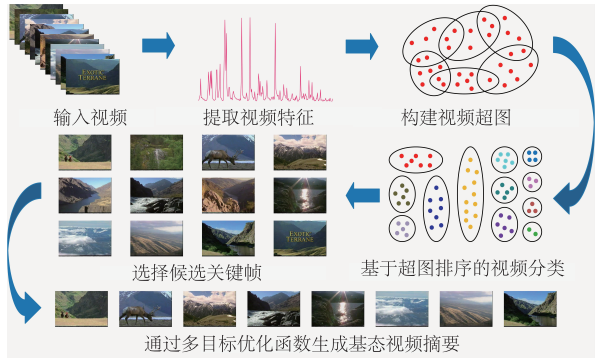

图1 所提HGRVS算法示意图

2 所提 HGRVS 算法

图 1 描述了所提 HGRVS 算法的整体框图. 对于输入的视频首先提取其视频帧特征,接着将这些特征作为超图的顶点来构建超图模型,然后通过本文提出的基于超图排序的视频帧分类算法将视频帧进行分类,最后将关键帧的提取视作一种目标函数的优化问题,通过求解生成视频摘要.

2.1 构建视频超图

超图是对简单图的扩展,简单图一条边只能包含 2 个顶点,而超图的超边可以包含任意多个顶点[21]. 超图通常使用关联矩阵 $\boldsymbol{H} = |\boldsymbol{V}| \cdot |\boldsymbol{E}|$ 来表示,定义如下[14]:

$$h(v_i, e_j) = \begin{cases} A(i,j), & \text{if } v_i \in e_j \\ 0, & \text{if } v_i \notin e_j \end{cases} \tag{1}$$

其中 $A(i,j) = \exp(-dis(v_i, v_j))$,$v_i$ 为属于超边 e_j 的任意顶点,v_j 为超边 e_j 的中心点,$dis(v_i, v_j)$ 为顶点 v_i 到超边中心点 v_j 归一化后的距离. 此外,超边的权重 $w(e_i)$ 定义为 $w(e_i) = \sum_{v_j \in e_i} A(i,j)$,超边的度定义为 $\sigma(e) = \sum_{v \in e} h(v,e)$,顶点的度定义为 $d(v) = \sum_{e \in E} w(e) \cdot h(v,$

e). 顶点的度、超边的度和超边权重构成的对角线矩阵分别称为顶点度矩阵 $\boldsymbol{D}_v$,超边度矩阵 $\boldsymbol{D}_e$,超边权重矩阵 $\boldsymbol{W}$.

本文使用基于 k 近邻的方法[14]构建视频超图,将视频特征看作超图的顶点,将每一个 $v_i \in \boldsymbol{V}$ 的顶点与其 k 近邻的顶点使用一条超边连接,并根据式(1)计算关联矩阵 $\boldsymbol{H}$.

2.2　基于超图排序的视频帧分类算法

在视频摘要技术中,相似视觉或语义内容只需使用一个关键帧来表示,因此可以将视频帧按照视觉或语义内容进行分类,然后从每个子类中选出关键帧.

本部分算法主要受两种思想启发,一种是超图排序算法,另一种是求取主集(Dominant Set)的过程. 超图排序(Hyper-Graph Ranking)算法是一种半监督的学习过程[15],其基本思想是将超图的一个或几个顶点作为标记顶点,其余顶点作为未标记顶点,利用超图的直推式学习方法将未标记顶点按照得分排序,得分值越高的顶点与标记顶点越相似. 该算法广泛应用于多媒体检索和查询领域[14,15]. 主集与聚类类似,每一个主集相当于聚类中的一类,同一个主集中的数据之间十分相似. Pavan 等人[22]提出了一种通过迭代求解二次方程将图划分为不同主集的方法,每一个主集内部之间的顶点比主集与主集之间的顶点有较强的联系,该算法的过程为首先计算出一个主集,然后从图中移除上次计算出的主集所包含的顶点,然后迭代上述过程直到发现所有主集. 随后,Liu 等人[23]提出了归一化主集(Normalized Dominant Set),对图的顶点引入权重,并进一步通过条件限制了主集之间的顶点保持强连接,生成归一化主集的过程与文献[22]的过程类似. 主集在物体追踪[24]、图像分割[25]等领域有着广泛的应用.

将以上两种思想相结合,如果把视频帧中的一帧作为超图的标记顶点,其余视频帧作为超图的未标记顶点,利用超图排序算法则可以计算出未标记视频帧相对于标记视频帧的得分排序. 此时可将得分值大于某个阈值的视频帧看作与标记视频帧内容相似的同类帧,以此循环则可将视频帧进行分类. 基于此,本文提出了一种利用超图排序算法对视频帧分类的算法,具体如下.

对于带权重的概率超图 $G=(V,E,w)$,定义一个 $\boldsymbol{f} \in R^{|V|}$ 的得分向量 $\boldsymbol{f}$,使得每一个代表视频帧的超图顶点 v 都有一个对应的得分 $f(v)$. 为了使同一条超边内的顶点具有较大的相似性,不同超边内的顶点相似性较小[21],定义目标函数 $\Omega(\boldsymbol{f})$ 为:

$$\Omega(\boldsymbol{f}) = \frac{1}{2}\sum_{e\in E}\sum_{\{u,v\in e\}}\frac{w(e)}{\sigma(e)}\left(\frac{\boldsymbol{f}(u)}{\sqrt{d(u)}}-\frac{\boldsymbol{f}(v)}{\sqrt{d(v)}}\right)^2 \tag{2}$$

其中顶点 u 和 v 属于同一条超边 e,$\sigma(e)$ 和 $w(e)$ 分别为超边 e 的度和权重,$d(u)$ 和 $d(v)$ 分别为顶点 u 和 v 的度. 为了使同一条超边内顶点的得分值尽可能接近,函数 $\Omega(\boldsymbol{f})$ 的值越小越好.

定义矩阵 $\boldsymbol{\Theta}=D_v^{-1/2}\boldsymbol{H}\boldsymbol{W}\boldsymbol{D}_e^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{D}_v^{-1/2}$,根据式(2)可以推导出:

$$\Omega(\boldsymbol{f}) = \boldsymbol{f}^{\mathrm{T}}(\boldsymbol{I}-\boldsymbol{\Theta})\boldsymbol{f} = \boldsymbol{f}^{\mathrm{T}}\Delta\boldsymbol{f} \tag{3}$$

其中 $\boldsymbol{I}$ 代表单位矩阵,$\Delta=\boldsymbol{I}-\boldsymbol{\Theta}$ 称为超图的拉普拉斯矩阵.

此外,如果给定一个初始化标签向量 $\boldsymbol{y}$,则得分向量 $\boldsymbol{f}$ 与 y 的值越接近越好,因此定义损失函数 $R(\boldsymbol{f})$ 为:

$$R(\boldsymbol{f}) = \|\boldsymbol{f}-\boldsymbol{y}\|^2 = \sum_{u\in V}(\boldsymbol{f}(u)-\boldsymbol{y}(u))^2 \tag{4}$$

因此得分向量 $\boldsymbol{f}$ 可通过下面的最优化问题来求解:

$$\underset{f\in R^{|V|}}{\operatorname{argmin}}\{\Omega(\boldsymbol{f})+\mu R(\boldsymbol{f})\} = \underset{f\in R^{|V|}}{\operatorname{argmin}}\{\boldsymbol{f}^{\mathrm{T}}(\boldsymbol{I}-\boldsymbol{\Theta})\boldsymbol{f}+\mu\|\boldsymbol{f}-\boldsymbol{y}\|^2\} \tag{5}$$

其中 μ 为平衡前后两项的参数,$\mu>0$. 由式(5)解得:

$$\boldsymbol{f} = (1-\lambda)(\boldsymbol{I}-\lambda\boldsymbol{\Theta})^{-1}\boldsymbol{y} \tag{6}$$

其中 $\lambda = 1/(1+\mu)$.

利用式(6)可以将视频帧进行分类. 首先在未分类的视频帧中随机选取一帧作为标记帧,对应的超图顶点为标记顶点. 设置初始化标签向量 $\boldsymbol{y}$,标记帧记为 1,未标记视频帧记为 0. 根据式(6)计算出所有未标记视频帧相对于标记帧的得分向量 $\boldsymbol{f}$. 设置一个最小得分阈值 η,认为 $\boldsymbol{f}>\eta$ 的视频帧与标记帧内容相似. 记 $\boldsymbol{f}>\eta$ 的视频帧个数为 N_1,为了使 N_1 个视频帧代表足够多的视频信息,设定阈值 T_1. 如果 N_1 小于 T_1,认为 N_1 个视频帧过短且没有代表足够多的视频内容,将重新随机选取一帧作为标记帧. 如果 N_1 大于 T_1,为了保证与标记帧同类的视频帧能够被正确分类,进一步引入再确认机制. 即从 N_1 个视频帧中随机选取几帧同样作为标记帧,将标签向量 $\boldsymbol{y}$ 的相应位置置为 1,然后根据式(6)重新计算 $\boldsymbol{f}>\eta$ 的视频帧范围及个数 N_2. 此时将 N_2 个视频帧作为 Cluster 1. 记未分类视频帧个数为 Nu,设截止阈值为 T_2,如果 Nu 小于 T_2 时,剩余未分类视频帧将不进行新的分类,终止算法. 如果 Nu 大于阈值 T_2 时,在剩余未分类的视频帧中重复上述过程. 最终可得到视频帧的 n 个分类 Cluster1 到 Cluster n. 上述过程如算法 1 所示. 图 2 是视频帧分类示意图.

算法 1　基于超图排序的视频帧分类算法

输入:视频帧序列

输出:视频帧的 n 个分类 $A=\{A_1,A_2,\cdots,A_n\}$

过程:

1:　提取视频帧特征.

2:　根据 k 近邻方法构建视频超图 G,并计算超图的拉普拉斯矩阵 $\boldsymbol{\Delta}$,$\boldsymbol{\Delta}=\boldsymbol{I}-\boldsymbol{\Theta}$.

3： 记输出类别号 $j=1$.

4： 在未分类的视频帧中随机选取第 i 帧作为标记帧,对应的视频超图顶点为 v_i,设置初始化标签向量 $\boldsymbol{y}=[0,0,\cdots,1,\cdots,0]^{\mathrm{T}}$,其中 $y_i=1$. 根据式(6)计算其余视频帧的得分向量 $\boldsymbol{f}$,统计得分向量 $\boldsymbol{f}>\eta$ 的未分类视频帧个数 N_1.

5： 若 $N_1<T_1$,认为步骤 4 选择的标记帧不具有代表性,返回步骤 4 重新选择.

6： 若 $N_1\geqslant T_1$,则从 N_1 个视频帧中选取若干帧同样作为标记帧,重置标签向量 y,再次根据式(6)计算视频帧得分向量 $\boldsymbol{f}$,统计得分向量 $\boldsymbol{f}>\eta$ 的未分类视频帧个数 N_2. 将 N_2 个视频帧作为同类,得到 A_j. 令 $j=j+1$.

7： 统计未分类视频帧个数 Nu,若 $N_u>T_2$,返回步骤 4,若 $Nu\leqslant T_2$,算法结束.

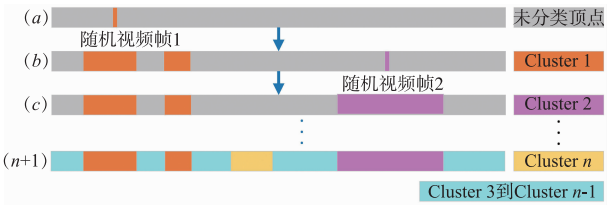

图2 视频帧分类示意图(a)在未分类视频中随机选取视频帧1作为标记帧,可得到(b)中Clusterl;(b)在剩余未分类视频帧中随机选取视频帧2作为标记帧,可得到(c)中Cluster 2;重复此过程直到将所有视频帧分类完毕。得到最终分类结果Cluster 1到Cluster n

2.3 关键帧的确定

算法 1 将视频帧分为 n 个类 $A=\{A_1,A_2,\cdots,A_n\}$,接下来只需要从 A 的各个类别中选择具有代表性的视频帧作为最终的视频摘要. 为此,本文采用一种两步法来确定视频摘要的关键帧,即首先从视频帧的 n 个分类中选取候选关键帧,再从候选关键帧中选取最终的关键帧构成视频摘要.

2.3.1 候选关键帧的选择

同一个类别 A_i 中的视频帧具有相似的视觉或语义内容,因此首先从每类中选取一帧作为候选关键帧. 其规则为从每类中选取距离该类平均特征向量距离最近的一帧作为候选关键帧,即:

$$c_i=\underset{a\in A_i}{\operatorname{argmin}}\left\|a-\bar{a}_i\right\|_2 \tag{7}$$

其中 $\bar{a}_i$ 为 A_i 中视频帧特征向量的平均值,$\boldsymbol{a}$ 为 A_i 中任意一个视频帧的特征向量. 由此可生成候选关键帧集合 C,$C=\{c_1,\cdots,c_i,\cdots,c_n\}$.

2.3.2 视频摘要的生成

此时候选关键帧集合可能有冗余帧的出现,因此需要进一步去除冗余帧得到最终摘要. 一个好的视频摘要通常要满足以下 3 个条件[26]:(1)最大不相关性,即关键帧之间要尽可能的不相似并保证多样性,尽可能代表不同的视频内容.(2)最大信息覆盖性,即选取的关键帧集合要代表尽可能多的视频内容,这样有利于对视频的整体理解.(3)最小信息冗余性,即关键帧集合中不能出现重复或相似的视频帧或对理解视频内容无帮助的视频帧,简洁无冗余的视频摘要不仅能方便浏览,还能降低对存储的要求. 通常,视频摘要越长越容易满足最大信息覆盖性,而视频摘要越短越容易保证最小信息冗余性,因此通常需要在两者之间做出平衡. 为了同时满足上面 3 个条件,本文设计了如下目标函数:

$$\begin{aligned}&S_1=\max\{I_i\}\\&S_{k+1}=S_k\cup\underset{c_i'\in C^c}{\operatorname{argmax}}\{\sigma Sim(c_i',S_k)+(1-\sigma)I_i\}\\&\text{s.t.}\ \sum I_{k+1}>\beta\end{aligned} \tag{8}$$

其中 S_{k+1} 为最终的视频摘要集合. $I_i=m_i/\sum_{i=1}^{n}m_i$ 为候选关键帧 c_i 代表的视频信息量,其中 m_i 为 c_i 所在分类的视频帧个数. 候选关键帧所在分类中的视频帧越多,候选关键帧代表的视频信息量越大. $Sim(c_i',S_k)=\min\limits_{s\in S_k,c_i'\in C^c}Sim(c_i',s)$,其中 c_i' 为剩余的候选关键帧,$c_i'\in C^c$,C^c 为已确定关键帧集合 S_k 在候选关键帧集合 C 中的补集,即 $C^c=C\backslash S_k$,s 为已选的关键帧,$s\in S_k$. $Sim(c_i',s)$ 即 c_i' 与 s 的距离,定义为 $\|c_i'-s\|_2$,因此该项可以从 C^c 中选取距离 S_k 最远的一帧作为关键帧,能够保证关键帧之间尽可能的不相似,保证了关键帧之间的最大不相关性. σ 为用来平衡前后两项重要性的参数,$0\leqslant\sigma\leqslant1$. $\sigma=0$ 时完全根据候选关键帧所代表的信息大小生成视频摘要,$\sigma=1$ 时完全依据关键帧之间最大不相关性生成视频摘要. $\sum I_{k+1}>\beta$ 为收敛条件,其中 β 为收敛阈值,$0<\beta\leqslant1$,控制关键帧集合 S_{k+1} 代表视频的总信息量,决定视频摘要的长度. 因此,通过控制 β 的值,可以控制视频摘要的最大信息覆盖性,还可以控制视频摘要的最小信息冗余性. 综上可知,关键帧的选取过程满足了以上 3 个条件.

生成最终视频摘要的整个过程如算法 2 所示.

算法 2 关键帧确定算法

输入:视频帧的 n 个分类 $A=\{A_1,A_2,\cdots,A_n\}$

输出:视频摘要集合 S_{k+1}

过程:

1： 根据式(7)计算候选关键帧集合 C,$C=\{c_1,\cdots,c_i,\cdots,c_n\}$.

2： 根据 $I_i=m_i/\sum_{i=1}^{n}m_i$ 计算每个候选关键帧代表视频的信息量 I_i.

3： 选取信息量 I_i 最大的候选关键帧作为第一个关键帧集合 S_1.

4： 按照式(8)迭代计算剩余关键帧,直到 $\sum I_{k+1}>\beta$. 得到视频摘要集合 S_{k+1},算法结束.

2.4 计算复杂度分析

HGRVS 算法计算复杂度由三部分组成,首先设超

图的顶点个数为 N,N 正比于视频的长度. 第一部分即 2.1 节使用 k 近邻法构建超图的过程,该过程首先需要计算顶点两两之间的相似性,其计算复杂度为 $O(N^2)$,然后需要计算出每个顶点的 k 近邻点连同该点作为一条超边,计算一个顶点的 k 近邻点时间复杂度为 $O(N\lg N)$,则计算出 N 个顶点的 k 近邻点的计算复杂度为 $O(N^2\lg N)$,因此该部分的计算复杂度为 $O(N^2\lg N)+O(N^2)$;第二部分即 2.2 节基于超图排序的视频帧分类算法,其计算复杂度主要由式(6)计算得分向量 f 产生,其中 Θ 矩阵的维度为 $N\times N$,则计算一次 f 复杂度为 $O(N^2)$,假设完成视频帧分类需要迭代 n 次,则此部分的计算复杂度为 $O(nN^2)$;第三部分即 2.3 节关键帧的确定算法,其时间主要由式(8)中 S_{k+1} 的迭代过程产生,每迭代一次需要计算 n 个候选关键帧与已选关键帧集合之间两两相似性,计算复杂度为 $O(n^2)$,设满足迭代条件 $\sum I_{k+1}>\beta$ 时迭代 m 次,则该部分计算复杂度为 $O(mn^2)$. 因此 HGRVS 算法总的计算复杂度为三部分计算复杂度之和,即 $O(N^2\lg N)+O(N^2)+O(nN^2)+O(mn^2)$,当 $k<n\leqslant N$ 时,HGRVS 算法的计算复杂度为 $O(N^2\lg N)$.

3　实验结果及分析

实验选取两个数据集验证所提 HGRVS 算法的有效性:(1)Open Video Project 数据集. 使用 VSUMM[3] 收集的 50 个来自 Open Video Project 的视频,视频均为 MPEG-1 格式,长度为 1 ~ 4 分钟. 视频内容分为纪录片、教育、演讲、历史等多个主题. 该数据集提供了 5 个用户手工选取的摘要作为客观评价标准的基准,因此在视频摘要领域中被广泛使用[5,11,12]. (2)YouTube 数据集. 该数据集由自行从 YouTube 视频网站上下载的 20 个视频组成,视频长度均为 1 ~ 4 分钟. 视频内容分为电影片段、监控视频、BBC 新闻、体育比赛 4 个主题.

3.1　实验设置

3.1.1　视频特征提取

视频帧是组成视频的最基本元素. 对于输入的视频,首先对视频进行每秒 2 帧的采样,将视频特征的提取转化为对采样视频帧特征的提取. 由于颜色特征提取简单,且对相机拍摄位置的变化有较强的鲁棒性,因此常用在视频摘要领域. 本文采用文献[27]中的方式提取 HSV 颜色特征. HSV 颜色空间被分为 256 个子空间,其中 H 空间被分为 16 个子空间、S 空间被分为 4 个子空间、V 空间被分为 4 个子空间.

3.1.2　实验参数分析及设置

本文提出的 HGRVS 算法主要受 4 个参数的影响. (1)使用 k 近邻方法构建超图时 k 值的选择. k 值越大,超图的超边包含的视频帧越多,这时更多的视频帧将建立内在的联系. 较大的 k 值将导致原本没有关系的视频帧建立联系,这就可能导致在视频帧进行分类时使原本不是同一类的视频帧划分为同一类. 反之,较小的 k 值将导致原本存在关系的视频帧失去内在的联系,这就可能导致在视频帧进行分类时使原本是同一类的视频帧划分为多个类. 因此 k 值的选择直接影响视频帧分类结果的好坏. (2)基于超图排序的视频帧分类算法中得分阈值 η 的选择. 通常与标注视频帧不是同一个分类的视频帧的得分值和与标注视频帧是同一个分类的视频帧的得分值数量级差别明显,因此 η 可以设置为一个较小的常数. (3)式(8)中 σ 值的选择. 不同的 σ 值将按照不同的原则生成视频摘要. (4)控制式(8)收敛的阈值 β 的选择. β 值的大小将直接影响视频摘要的长度. 另外影响实验结果的参数还有算法 1 中的阈值 T_1 和 T_2,这两个参数对实验结果影响较小,通常设置为一个较小的常数即可. 本文中的实验参数分别设置为 $k=10$、$\eta=0.001$、$\sigma=0.7$、$\beta=0.8$、$T_1=5$、$T_2=10$.

3.1.3　对比算法

本文分别与等间隔采样(Uniform Sampling,简称为 US)、k 均值聚类(k-means)、OV[28]、DT[4]、STIMO[2]、Kuanar 的方法[5]等算法进行比较. 其中,US 是间隔相同的时间对视频采样,并将采样结果作为最后的视频摘要结果. 设关键帧个数为 m,视频长度为 d,采样时间计算公式为 $t_i=(d/m)(1/2+i)$,其中 $i\in[0,\cdots,m-1]$. k 均值聚类算法是一种常用的聚类方法,本文选取距离聚类中心点最近的视频帧作为关键帧. 由于聚类结果受初始聚类中心选择影响较大,因此本文随机实验 100 次求取平均值. OV、DT、STIMO 的实验结果可以从文献[3]获取. Kuanar 的方法的实验分别使用了颜色特征和颜色特征加边缘特征,因此实验结果分为两种,分别记为 KMC(Kuanar's Method + C)和 KMCE(Kuanar's Method + CE),Kuanar 的方法的实验结果可从文献[5]获取.

3.1.4　评价标准的选择

视频摘要评价标准通常分为三类[29]:(1)结果描述,一般只需要分析实验参数对实验结果的影响,不需要与其他方法进行比较,是一种最简单的视频摘要评价标准. (2)客观评价,通常使用评价函数来评定视频摘要结果的好坏. 常用的评价函数有镜头重构度(SDR)[30],准确率(CUS_A)和误差率(CUS_E)[3],精度(Precision)、召回率(Recall)和 F 值(F-score)[11]等. (3)主观评价,通过用户来主观判断视频摘要的好坏,是最直接和最有效的一种评价标准. 一般通过用户打分[7]、摘要结果评级[31]等方式进行评价. 在 Open Video Project 数据集上,本文将利用客观评价和主观评价两种方式进行评价. 在 YouTube 数据集上,由于没有通用的

视频摘要基准,因此仅使用主观评价方式进行评价.

3.2 Open Video Project 数据集结果分析

3.2.1 客观评价标准结果及分析

本文使用两套主流的客观评价标准进行评价,一套是精度(Precision)、召回率(Recall)和 F 值(F-score),另一套是准确率(CUS_A)和误差率(CUS_E).上述两套标准的评测均需用到数据集 VSUMM 提供的用户摘要标准,而其标准来自 5 个用户,因此本文对所有实验结果求取平均值.

(1) 精度、召回率和 F 值结果对比及分析

精度、召回率和 F 值的计算公式分别如下:

$$精度(\text{Precision}) = \frac{N_{\text{matched}}}{N_{\text{AS}}} \tag{9}$$

$$召回率(\text{Recall}) = \frac{N_{\text{matched}}}{N_{\text{US}}} \tag{10}$$

$$F\ 值 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{11}$$

其中 N_{matched} 表示自动摘要与用户摘要匹配的长度,即自动摘要中与用户摘要中相同的关键帧个数,定义当两个关键帧的颜色直方图的曼哈顿距离小于指定阈值 φ 时,认为两个关键帧是匹配的,本实验将阈值 φ 设置为 0.5;N_{AS} 表示自动生成摘要的长度;N_{US} 表示用户摘要长度.精度反映了自动摘要摘选出匹配关键帧的能力,召回率反映了匹配关键帧击中用户摘要的能力,F 值是对精度和召回率的平衡,是对视频摘要好坏的一个整体评价.表 1 为不同算法的实验结果数据对比.

表 1 Open Video Project 数据集上不同算法的精度、召回率和 F 值结果对比

算法	US	k-means	OV[28]	DT[4]	STIMO[2]	KMC[5]	KMCE[5]	**HGRVS**
精度	0.55	0.54	0.62	0.61	0.56	0.68	**0.70**	0.69
召回率	0.69	0.69	0.69	0.48	0.67	0.59	0.57	**0.70**
F-值	0.61	0.61	0.65	0.54	0.61	0.63	0.63	**0.70**
平均摘要长度	10.00	10.00	9.66	6.14	9.96	6.94	6.62	8.76

由表 1 可以看出,所提 HGRVS 算法的 F 值在所有算法中最高,分别比 US、k-means、OV、DT、STIMO、KMC、KMCE 算法高 9%、9%、5%、16%、9%、7%、7%,说明本文的 HGRVS 算法的整体性能高于其他算法.具体来看,所提 HGRVS 算法的精度略低于 KMCE 算法,分别比 US、k-means、OV、DT、STIMO、KMC 算法高 14%、15%、7%、8%、13%、1%.而所提 HGRVS 算法的召回率在所有算法中最高.可以发现 US、k-means、STIMO 几种方法的召回率都比较高,但精度却比较低,这是由于这几种算法在提高自动摘要与用户摘要匹配长度的同时,导致平均摘要长度都比较长,从而有较高的召回率和较低的精度.另外,所提 HGRVS 算法的平均长度也较为适中.综合来看,本文提出的 HGRVS 算法在精度、召回率和 F 值这套性能评测标准中具有最好的表现.

(2) 准确率和误差率结果对比及分析

准确率和误差率的计算公式如下:

$$准确率(CUS_A) = \frac{N_{m\text{AS}}}{N_{\text{US}}} \tag{12}$$

$$误差率(CUS_E) = \frac{N_{\bar{m}\text{AS}}}{N_{\text{US}}} \tag{13}$$

其中 N_{mAS} 表示自动摘要与用户摘要匹配的长度,$N_{\bar{m}AS}$ 表示自动摘要与用户摘要未匹配的长度,即自动摘要长度减去匹配长度,N_{US} 表示用户摘要长度.准确率越大越好,误差率越小越好.表 2 给出了所提算法与对比算法的实验数据.

从表 2 中可以看出,HGRVS 算法的准确率与 US、k-means、OV 和 STIMO 算法类似,但是误差率值远小于这些算法,可见 HGRVS 算法优于这些算法.另外,HGRVS 算法的准确率值高于 DT、KMC 和 KMCE 算法,但是误差率的值也较高于这些算法,因此不宜进行直观比较.为此,通过进一步实验得到了准确率和误差率随 k 值的变化曲线图,如图 3 所示.可知,当 $k=15$ 时,HGRVS 算法的误差率为 0.32,与 KMC 和 DT 算法接近,但此时准确率为 0.62,高于这两种算法.当 $k=17$ 时,HGRVS 算法的误差率为 0.29,与 KMCE 算法相同,但准确率为 0.59,高于 KMCE 算法.可见,本文所提 HGRVS 算法在准确率和误差率这套性能评测标准中也好于其他算法.

3.2.2 主观评价结果及分析

主观评价采用摘要结果评级的方式,使用 Good、Acceptable 和 Bad 三个等级对视频摘要结果进行评定[31].邀请与本研究无关的 2 男 3 女作为测试者,通过

表 2 Open Video Project 数据集上不同算法的准确率和误差率结果对比

算法	US	k-means	OV[28]	DT[4]	STIMO[2]	KMC[5]	KMCE[5]	HGRVS
准确率	0.69	0.69	0.69	0.48	0.68	0.59	0.57	0.69
误差率	0.72	0.71	0.55	0.33	0.63	0.32	0.29	0.39

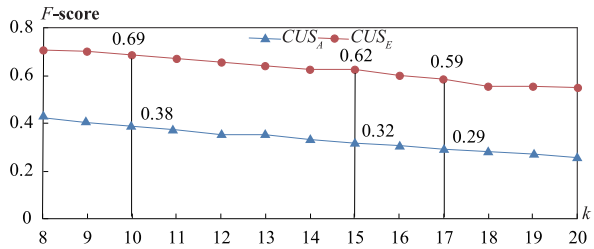

图3　准确率和误差率随k值的变化曲线图

先观看视频摘要结果,再观看原有视频,然后将关键帧与原始视频的镜头进行对比,看关键帧能够代表镜头的个数,并对本文摘要结果的满意度进行评定.

本文对测试条件进行如下约定:对于时间短于3s的镜头可认为是无意义且无需使用关键帧来表示的镜头,对于重复出现的镜头可使用一个关键帧来表示. 主观评价结果见表3.

表3　Open Video Project 数据集主观评价结果

	Good	Acceptable	Bad	总数
用户1	42(82%)	7(14%)	1(4%)	50
用户2	43(86%)	7(14%)	0(0%)	50
用户3	42(84%)	6(12%)	2(4%)	50
用户4	43(86%)	7(14%)	0(0%)	50
用户5	44(88%)	6(12%)	0(0%)	50
平均值	42.8(85.6%)	6.6(13.2%)	0.6(1.2%)	50

从表3中可以看出,等级 Good 的百分比占到85.6%,说明 HGRVS 算法对于多数视频生成的摘要结果能够很好地呈现视频内容. 等级 Acceptable 的百分比占到13.2%,说明 HGRVS 算法对于部分视频生成的摘要结果能够较好地呈现视频内容. 等级 Bad 的百分比仅有1.2%,说明 HGRVS 算法对少数的视频生成的摘要结果不能够呈现其视频内容. 总体来看,HGRVS 算法生成的摘要能够较好地反应视频的主要内容.

3.2.3　算法鲁棒性分析

为进一步证明所提 HGRVS 算法对不同特征的鲁棒性,分别使用了颜色特征、边缘特征[5]、和 C3D 特征[32]生成视频摘要,并计算各自 F 值. 本文采用文献[5]中方式提取边缘特征,首先将视频帧划分为4×4的子块,然后分别使用水平、垂直、45度、135度和无方向性的检测算子计算子块的边缘特征,最终得到80维的边缘特征. C3D(Convolution 3D feature)特征[32]是最近提出的一种基于卷积神经网络(CNN)的视频特征,相对于颜色和边缘特征是一种高层特征,在目标识别、场景分类等领域都有较好的应用.

图4、图5和图6是分别使用颜色特征、边缘特征和C3D特征的视频摘要结果的 F 值曲线对比图,三幅图均为固定 σ、β 和 k 其中两个参数,改变另一个参数所得到. 从三幅图中可见,对于固定其中两个参数,改变另一个参数时,3种特征下 F 值十分接近,说明 HGRVS 算法对不同特征具有较好的鲁棒性. 从图4中可以看出,随着 k 的增加,三种特征下的 F 值均先变大在变小,说明构建超图时只有选择合适的 k 值才能获得较好的视频摘要结果. 从图5可以发现,随着 σ 的增大,三种特征下的 F 值均逐渐变大,说明按照最大不相关性更容易获得较好的视频摘要结果. 图6中随着 β 的增大,三种特征下的 F 值同样逐渐变大,当 β 等于0.8时趋于平稳,说明视频摘要结果包含的信息量为0.8时即可获得较好的视频摘要结果.

在三幅对比图中,颜色特征总是具有相对较好的性能是因为本文固定的两个参数总是在颜色特征下调参时的最优结果.

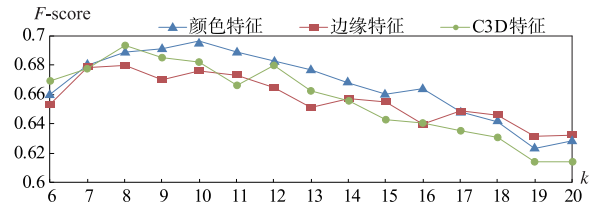

图4　固定参数σ与β，改变参数k时不同特征下的F值变化曲线图

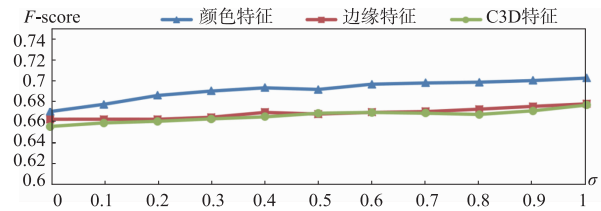

图5　固定参数k与β，改变参数σ时不同特征下的F值变化曲线图

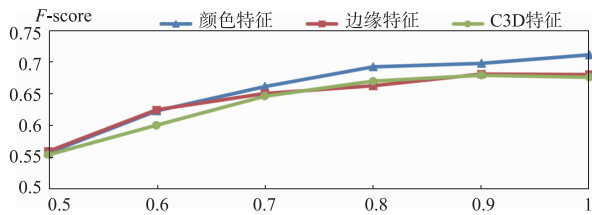

图6　固定参数k与σ，改变参数β时不同特征下的F值变化曲线图

3.3　YouTube 数据集结果分析

在本数据集上,由于没有通用的摘要标准结果,因而只使用主观评价进行评定. 采用与 Open Video Project 数据集上相同的主观评价方法,主观评价结果见表4.

从表4可以看出,在 YouTube 数据集上,HGRVS 算法对多数视频能够较好地呈现其内容,只有对少数视频表现不尽人意. 与 Open Video Project 数据集上的结果对比可知,YouTube 数据集上等级为 Good 的百分比

略低,主要原因是从 YouTube 视频网站收集的视频的画面运动激烈程度以及镜头切换速度都要高于 Open Video Project 数据集上的视频. 另外两个数据集上等级为 Good 和等级为 Acceptable 的百分比之和几乎相等,说明对于不同的视频 HGRVS 算法生成的视频摘要结果整体上能较好地呈现出视频内容.

表 4 YouTube 数据集主观评价结果

	Good	Acceptable	Bad	总数
用户 1	15(75%)	4(20%)	1(5%)	20
用户 2	17(85%)	3(15%)	0(0%)	20
用户 3	16(80%)	3(15%)	1(5%)	20
用户 4	16(80%)	4(20%)	0(0%)	20
用户 5	18(90%)	2(10%)	0(0%)	20
平均值	16.4(82%)	3.2(16%)	2(2%)	20

4 总结与展望

本文通过将超图模型引入到视频摘要领域,提出了一种新颖的静态视频摘要方法,称作基于超图排序的视频摘要算法(HGRVS). 该方法首先利用视频超图模型捕获视频帧之间的复杂关系,利用超图排序的思想将视频帧进行分类;然后从视频帧分类中选取候选关键帧,最后通过求解目标函数生成静态视频摘要. 通过在 Open Video Project 和 YouTube 两个数据集上的主观与客观的详细实验,以及与当前主流的视频摘要方法的对比,验证了本文所提算法的有效性及先进性.

接下来将从两方面开展后续研究工作,(1)关于对 HGRVS 进行加速,本文的超图模型相比于图模型规模更大,计算更耗时,因此考虑使用哈希等算法进行加速[33,34];(2)基于 k 近邻法构建视频超图时 k 值的选择进行研究,不同的视频往往需要选择不同的 k 值使得摘要结果最优,因此考虑设计一种针对不同视频的自适应选择 k 值的方法来构建视频超图,使得在每个视频的摘要结果都是最优,从而使整体性能也达到最优.

参考文献

[1] Herranz L, Martínez JM. An efficient summarization algorithm based on clustering and bitstream extraction[A]. Proceedings of the IEEE International Conference on Multimedia and Expo[C]. New York: IEEE, 2009. 654 – 657.

[2] Furini M, Geraci F, Montangero M, et al. STIMO: Still and moving video storyboard for the web scenario[J]. Multimedia Tools and Applications, 2010, 46(1): 47 – 69.

[3] DeAvila SEF, Lopes APB. VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method[J]. Pattern Recognition Letters, 2011, 32(1): 56 – 68.

[4] Mundur P, Rao Y, Yesha Y. Keyframe-based video summarization using delaunay clustering[J]. International Journal on Digital Libraries, 2006, 6(2): 219 – 232.

[5] Kuanar SK, Panda R, Chowdhury AS. Video key frame extraction through dynamic delaunay clustering with a structural constraint[J]. Journal of Visual Communication and Image Representation, 2013, 24(7): 1212 – 1227.

[6] Zhai SL, Luo B, Tang J, et al. Video abstraction based on relational graphs[A]. Proceedings of the Fourth International Conference on Image and Graphics[C]. Sichuan: ACM, 2007. 827 – 832.

[7] Ngo CW, Ma YF, Zhang HJ. Automatic video summarization by graph modeling[A]. Proceedings of the Ninth IEEE International Conference on Computer Vision[C]. Nice: IEEE, 2003. 104 – 109.

[8] Ngo CW, Ma YF, Zhang HJ. Video summarization and scene detection by graph modeling[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2005, 15(2): 296 – 305.

[9] Lu S, King I, Lyu MR. Video summarization by video structure analysis and graph optimization[A]. Proceedings of the IEEE International Conference on Multimedia and Expo[C]. Taipei: IEEE 2004. 1959 – 1962.

[10] Panda R, Kuanar SK, Chowdhury AS. Scalable video summarization using skeleton graph and random walk[A]. Proceedings of the International Conference on Pattern Recognition[C]. Stockholm: IEEE, 2014. 3481 – 3486.

[11] Mei S, Guan G, Wang Z. Video summarization via minimum sparse reconstruction[J]. Pattern Recognition, 2015, 48(2): 522 – 533.

[12] Dang C, Radha H. RPCA-KFE: Key frame extraction for video using robust principal component analysis[J]. IEEE Transactions on Image Processing, 2015, 24(11): 3742 – 3753.

[13] 徐超,聂勇伟,葛红美,等. 基于新视角合成的视频摘要交互式浏览[J]. 电子学报,2015,43(11):2263 – 2270.
Xu Chao, Nie Yong-wei, Ge Hong-mei, et al. Novel-View synthesis based interactive video synopsis browsing[J]. Acta Electronica Sinica, 2015, 43(11): 2263 – 2270. (in Chinese)

[14] Huang YC, Liu QS, Hang ST, et al. Image retrieval via probabilistic hypergraph ranking[A]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition[C]. San Francisco: IEEE, 2010. 3376 – 3383.

[15] Wang W, Li SJ. Exploring hypergraph-based semi-supervised ranking for query-oriented summarization[J]. Information Sciences, 2013, 237(13): 271 – 286.

[16] Yang X, Wang M, Tao DC. Robust visual tracking via

multi-graph ranking [J]. Neurocomputing, 2015, 159(C):35 – 43.
[17] Lu RT, Xu WY, Zheng YB, et al. Visual tracking via probabilistic hypergraph ranking [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2015, (99):1 – 14.
[18] Deng C, Ji RR, Liu W, et al. Visual reranking through weakly supervised multi-graph learning [A]. Proceedings of the IEEE International Conference on Computer Vision [C]. Sydney: IEEE, 2013. 2600 – 2607.
[19] Deng C, Ji RR, Tao DC, et al. Weakly supervised multi-graph learning for robust image reranking [J]. IEEE Transactions on Multimedia, 2014, 16(16):785 – 795.
[20] Li L, Li T. News recommendation via hypergraph learning: encapsulation of user behavior and news content [A]. Proceedings of the ACM International Conference on Web Search and Data Mining [C]. New York: ACM, 2013. 305 – 314.
[21] Zhou DY, Huang JY, Schokopf B. Learning with hypergraphs: clustering, classification, and embedding [A]. Advances in Neural Information Processing Systems [C]. cambridge: NIPS, 2007. 1601 – 1608.
[22] Pavan M, Pelillo M. Dominant sets and pairwise clustering [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(1):167 – 172.
[23] Liu XL, He JF, Lang B, et al. Hash bit selection: A unified solution for selection problems in hashing [A]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition [C]. Portland: IEEE, 2013. 1570 – 1577.
[24] Tesfaye YT, Zemene E, Pelillo M, et al. Multi-object tracking using dominant sets [J]. IET Computer Vision, 2016, 10(4):289 – 297.
[25] Zemene E, Pelillo M. Interactive image segmentation using constrained dominant sets [A]. Proceedings of the European Conference on Computer Vision [C]. Amsterdam: Springer, 2016. 278 – 294.
[26] Fu YW, Guo YW, Zhu YS, et al. Multi-view video summarization [J]. IEEE Transactions on Multimedia, 2010, 12(7):717 – 729.
[27] Paschos G. Perceptually uniform color spaces for color texture analysis: an empirical evaluation [J]. IEEE Transactions on Image Processing, 2001, 10(6):932 – 937.
[28] Doermann D, Kobla V. Video summarization by curve simplification [A]. Proceedings of the ACM International Conference on Multimedia [C]. New York: ACM, 1998. 211 – 218.
[29] Truong BT, Venkatesh S. Video abstraction: a systematic review and classification [J]. Transactions on Multimedia Computing Communications and Applications, ACM, 2007, 3(1):1 – 37.
[30] Liu TY, Zhang X, Feng J, et al. Shot reconstruction degree: a novel criterion for key frame selection [J]. Pattern Recognition Letters, 2004, 25(12):1451 – 1457.
[31] Liu T, Zhang HJ, Qi F. A novel video key-frame extraction algorithm based on perceived motion energy model [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2003, 13(10):1006 – 1013.
[32] Du Tran, Lubomir Bourdev, Rob Fergus, et al. Learning spatiotemporal features with 3D convolutional networks [A]. Proceedings of the IEEE International Conference on Computer Vision [C]. Santiago: IEEE, 2015. 4489 – 4497.
[33] Deng C, Deng H, Liu XL, et al. Adaptive multi-bit quantization for hashing [J]. Neurocomputing, 2015, 151: 319 – 326.
[34] Liu XL, Deng C, Lang B, et al. Query-adaptive reciprocal hash tables for nearest neighbor search [J]. IEEE Transactions on Image Processing, 2016, 25(2):907 – 919.

作者简介

冀　中(通信作者)　男,博士,现为天津大学电子信息工程学院副教授,博士生导师. 主要研究方向为特征学习,计算机视觉,多媒体分析和检索.
E-mail:jizhong@ tju. edu. cn

樊帅飞　男,1990 年生于河南安阳,现为天津大学电子信息工程学院硕士研究生,研究方向为视频摘要.
E-mail:fanshuaifei@ tju. edu. cn