

一种基于二分图最优匹配的镜头检索方法

彭宇新^{1,2}, Ngo Chong-Wah³, 肖建国^{1,2}

(1. 北京大学计算机科学技术研究所, 北京 100871; 2. 北京大学文字信息处理国家重点实验室, 北京 100871;
3. 香港城市大学电脑科学系, 香港九龙)

摘 要: 镜头检索是基于内容的视频检索的重要内容. 本文首次尝试将二分图的最优匹配用于镜头检索. 与现有方法相比, 本文提出的方法强调在一一对应的前提下, 全面客观地度量两个镜头的相似度. 把两个镜头的相似度度量建模为一个带权的二分图: 镜头中的每一帧看成二分图的一个结点, 两个镜头之间任意帧的相似值作为边的权值. 在一一对应的前提下, 利用最优匹配的 Kuhn-Munkres 算法求出该二分图的最大权, 以此作为两个镜头的相似度. 考虑到检索速度问题, 提出了两个改进算法. 实验对比结果证实了本文所提方法在镜头检索中的优异表现.

关键词: 基于内容的镜头检索; 二分图的最优匹配; Kuhn-Munkres 算法; 改进算法

中图分类号: TP301 **文献标识码:** A **文章编号:** 0372-2112 (2004) 07-1135-05

An Approach for Shot Retrieval by Optimal Matching in the Bipartite Graph

PENG Yu-xin^{1,2}, NGO Chong-Wah³, XIAO Jian-guo^{1,2}

(1. Institute of Computer Science and Technology, Peking University, Beijing 100871, China;

2. National Key Laboratory of Text Processing Technology, Beijing 100871, China;

3. Dept. of Computer Science, City University of Hong Kong, Kowloon, Hong Kong, China)

Abstract: Shot retrieval plays a critical role in content-based video retrieval. Motivated by the theory of optimal matching in bipartite graph, we propose a novel approach based on the Kuhn-Munkres algorithm for shot retrieval. In contrast to existing algorithms, the proposed approach emphasizes one-to-one mapping among frames between two shots for effective similarity measure. A weighted bipartite graph is constructed to model the similarity between two shots: every vertex in a bipartite graph represents one frame in a shot, and the weight of every edge represents the similarity value for a pair of frames between two shots. Then Kuhn-Munkres algorithm is employed to compute the maximum weight of a constructed bipartite graph as the similarity value between two shots by guaranteeing the one-to-one mapping among frames. To improve the speed efficiency, we also propose two improved algorithms. Experimental results indicate that the proposed approach achieves superior performance than some existing methods.

Key words: Content-based shot retrieval; optimal matching; Kuhn-Munkres algorithm; improved algorithm

1 引言

随着电视台视频节目的积累、网上数字视频的增加, 以及数字图书馆、视频点播、远程教学等大量的多媒体应用, 如何在海量视频中快速检索出所需要的资料显得至关重要. 传统的基于关键词描述的视频检索因为描述能力有限、主观性强、手工标注等原因, 已经不能满足海量视频检索的需求. 因此, 从 90 年代开始, 基于内容的视频分析和检索技术成为研究的热点问题, 多媒体内容描述接口的 MPEG-7 标准的制定也引起了人们广泛的关注. 目前, 大量的研究主要集中在视频内容的结构分析上, 如镜头的分割、关键帧的提取、场景的构造等, 而视频检索方面的研究则相对较少, 但这部分常常有着最广

泛的应用.

视频检索的一般方法是首先进行镜头边界检测^[1], 以镜头作为视频序列的基本结构单元和检索单元; 然后在每个镜头内部提取关键帧来表示该镜头的内容, 从关键帧提取出颜色、纹理和形状等低级特征, 用于镜头的检索. 这样, 就把基于内容的镜头检索转化为基于内容的图像检索来解决. 例如在文献[2, 3], 采用基于非监督聚类的方法提取一个关键帧用来表示该镜头, 在文献[4], 提出一种基于最近特征线 (nearest feature line, 简称 NFL) 的端点检测算法用于选取关键帧. 这类方法存在的问题是: 镜头是图像在时间上的连续序列, 没有对存在于视频中的时间信息和运动信息充分进行利用. 文献[5, 6]提出把一个镜头内的内容变化分解为几个内容一致的子单

元,称为子镜头(subshot),两个镜头的相似性通过对应子镜头的相似性计算得到.在文献[5],基于相机运动来提取子镜头,然后针对不同运动的子镜头,构造不同的关键帧表示,如静止(static)子镜头可以用一个关键帧来表示,扫描(pan,tilt)子镜头通过构造一个全景图(mosaic,panoramic image)来表示,变焦(zoom)子镜头可以用变焦之前和之后的两个关键帧来表示,最后两个镜头的相似度表示为它们所有关键帧相似度的最大值和第二大值的平均值($\text{Sim}(s_i, s_j) = \frac{1}{2} \{ M(s_i, s_j) + M(s_i, s_j) \}$, $M(s_i, s_j)$ 表示两个镜头 s_i 和 s_j 的第二大相似值).在文献[6],提出了构造主色直方图和空间结构直方图来提取和表示子镜头,两个镜头的相似度表示为它们子镜头相似度的最大值($\text{Sim}(a, b) = \max_{i,j} (\text{Sim}(a_i, b_j))$, $\text{Sim}(a_i, b_j)$ 表示镜头 a 中的子镜头 i 和镜头 b 中的子镜头 j 的相似度).文献[5,6]提取子镜头的方法很好考虑了镜头内容的冗余性,并对存在于视频中的运动信息和时间信息进行了充分探索,但最大值和第二大值的度量方法并不能全面反映两个镜头内部的相似程度.为了解决这个问题,文献[7]假设两个相似镜头内部的相似帧按照时间顺序对应相似,因此,提出用动态规划算法(dynamic programming)度量两个镜头相似帧的数目,以此判断两个镜头是否相似.它的问题在于:提出的假设并不成立,两个镜头的相似帧未必按照时间顺序对应相似.同时,对镜头内每一帧的比较,也使得检索速度比较慢.

为了解决上述这些问题,本文提出利用二分图的最优匹配来度量两个镜头的相似度.把两个镜头的相似度度量建模为一个带权的二分图:镜头中的每一帧看成二分图的一个结点,两个镜头之间任意帧的相似值作为边的权值.在一一对应的前提下,利用最优匹配的 Kuhn-Munkres 算法求出该二分图的最大权,以此作为两个镜头的相似度.文献[5]考虑了使用 HSV 颜色直方图来表示关键帧的缺点:如果两个关键帧有相似的颜色分布,即使它们的内容不一样,也会认为这两个关键帧相似,因此使用最大相似值和第二大相似值的平均值来加强算法的鲁棒性.与文献[5]相比,本文提出的方法强调在一一对应的前提下来度量两个镜头的相似性,这样可以更加有效地避免文献[5]中所提的低层颜色和上层语义内容不一致的矛盾,而且能够全面客观地度量两个镜头内部的相似程度.它不仅解决了文献[5,6]没有全面度量两个镜头相似度的问题,也解决了文献[7]时间顺序限制的问题.实际上,与文献[7]运用的动态规划算法(dynamic programming)相比,二分图最优匹配的优点在于不限制时间顺序的前提下,达到准确度量镜头相似度的目的.但它也存在与文献[7]同样的问题,即镜头内每一帧的匹配,使得检索速度比较慢.基于这种考虑,本文提出了两个改进算法:子镜头构造带权二分图和等间隔采样构造带权二分图.第3部分的对比实验结果证实了本文所提方法的有效性.

为了解决上述这些问题,本文提出利用二分图的最优匹配来度量两个镜头的相似度.把两个镜头的相似度度量建模为一个带权的二分图:镜头中的每一帧看成二分图的一个结点,两个镜头之间任意帧的相似值作为边的权值.在一一对应的前提下,利用最优匹配的 Kuhn-Munkres 算法求出该二分图的最大权,以此作为两个镜头的相似度.文献[5]考虑了使用 HSV 颜色直方图来表示关键帧的缺点:如果两个关键帧有相似的颜色分布,即使它们的内容不一样,也会认为这两个关键帧相似,因此使用最大相似值和第二大相似值的平均值来加强算法的鲁棒性.与文献[5]相比,本文提出的方法强调在一一对应的前提下来度量两个镜头的相似性,这样可以更加有效地避免文献[5]中所提的低层颜色和上层语义内容不一致的矛盾,而且能够全面客观地度量两个镜头内部的相似程度.它不仅解决了文献[5,6]没有全面度量两个镜头相似度的问题,也解决了文献[7]时间顺序限制的问题.实际上,与文献[7]运用的动态规划算法(dynamic programming)相比,二分图最优匹配的优点在于不限制时间顺序的前提下,达到准确度量镜头相似度的目的.但它也存在与文献[7]同样的问题,即镜头内每一帧的匹配,使得检索速度比较慢.基于这种考虑,本文提出了两个改进算法:子镜头构造带权二分图和等间隔采样构造带权二分图.第3部分的对比实验结果证实了本文所提方法的有效性.

2 利用二分图的最优匹配进行基于内容的镜头检索

2.1 基于最优匹配的 Kuhn-Munkres 算法的镜头检索

二分图的最优匹配是图论中的一个经典问题,它解决的

一个典型应用问题如下:假设有 n 个工作人员 x_1, x_2, \dots, x_n 安排做 m 项工作任务 y_1, y_2, \dots, y_m ,如图1所示,其中边 $e_{ij} = (x_i, y_j)$ 表示 x_i 可以从事 y_j , e_{ij} 的权值 w_{ij} 表示 x_i 从事 y_j 的效率,如果每个人最多从事其中一项工作,且每项工作只能由一人承担.问:怎样合理安排使总的工作效率最高?

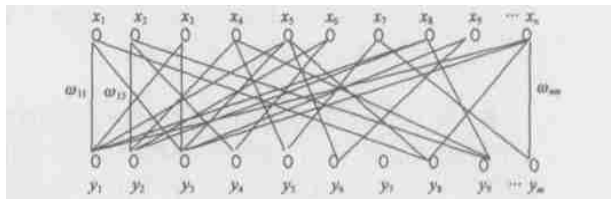


图1 最优匹配举例

把二分图的最优匹配运用到镜头检索中,可以这样表述镜头检索问题:设 X 镜头有 n 帧 x_1, x_2, \dots, x_n , Y 镜头有 m 帧 y_1, y_2, \dots, y_m ,其中边 $e_{ij} = (x_i, y_j)$ 表示 x_i 与 y_j 相似, w_{ij} 表示 x_i 与 y_j 的相似值,如图1所示.要求 x_i 与 y_j 必须一一对应而不能多对一或多对多对应,问:能使总的相似度最大的匹配?这样得到的匹配反映了 X 镜头和 Y 镜头在一一对应的条件下,所能达到的最大相似度,因此,本文把它作为这两个镜头的相似度.由这个表述,可以知道,最优匹配方法的实质是在一一对应的前提下,衡量两个镜头所能达到的最大相似度.因为不重复计算每一帧的相似度,而着眼于两个镜头所能达到的最大相似度,所以这个方法能够全面客观地衡量两个镜头内部的相似程度.

为了建模镜头相似度度量的带权二分图 $G = (X, Y, E)$ (其中,顶点集 $V = X \cup Y$,边集 $E = \{e_{ij}\}$),必须求出 x_i 和 y_j 的相似值 w_{ij} 作为每边 e_{ij} 的权值.本文使用直方图的交来计算

$$w_{ij} = \text{Intersect}(x_i, y_j) = \frac{1}{A(x_i, y_j)} \min_{h, s, v} \{ H_i(h, s, v), H_j(h, s, v) \} \quad (1)$$

$$A(x_i, y_j) = \min_{h, s, v} \{ H_i(h, s, v), H_j(h, s, v) \} \quad (2)$$

$H_i(h, s, v), H_j(h, s, v)$ 是 HSV 颜色空间的直方图,本文用 H,S,V 分量在 $18 \times 3 \times 3$ 的三维空间中统计直方图,以归一化后的 162 个数值作为颜色特征值. $\text{Intersect}(x_i, y_j)$ 表示两个直方图的交,用它来判断两个关键帧的相似性,使用 $A(x_i, y_j)$ 归一化 $\text{Intersect}(x_i, y_j)$ 到 0,1 之间.

Kuhn-Munkres 算法是解决上述最优匹配问题的经典算法.把两个镜头 X 和 Y 每对帧的相似值 w_{ij} 赋给 $G = (X, Y, E)$ 的每条边 e_{ij} ,这时的 G 就转化为一个带权的完全二分图(完全二分图表示 G 有 $m \times n$ 条边).

具体的 Kuhn-Munkres 算法^[8]如下:

$$(1) \text{ 给出初始标号 } l(x_i) = \max_j w_{ij}, l(y_j) = 0,$$

$$i, j = 1, 2, \dots, t, t = \max(n, m);$$

$$(2) \text{ 求出边集 } E_t = \{ (x_i, y_j) \mid l(x_i) + l(y_j) = w_{ij} \}, G_t = (X, Y, E_t) \text{ 及 } G_t \text{ 中的一个匹配 } M (M \subseteq E, \text{ 并且 } M \text{ 中任意两条边都不相邻});$$

(3) 如 M 已饱和 X 的所有结点, 则 M 即是 G 的最优匹配, 计算结束, 否则进行下一步;

(4) 在 X 中找一 M 非饱和点 x_0 , 令 $A \leftarrow \{x_0\}$, $B \leftarrow \emptyset$, A, B 是两个集合;

(5) 若 $N_{G_1}(A) = B$, 则转第(9)步, 否则进行下一步, 其中, $N_{G_1}(A) \subseteq Y_k$ 是与 A 中结点邻接的结点集合;

(6) 找一结点 $y \in N_{G_1}(A) - B$;

(7) 若 y 是 M 饱和点, 则找出 y 的配对点 z , 令 $A \leftarrow A \cup \{z\}$, $B \leftarrow B \cup \{y\}$ 转第(5)步, 否则进行下一步;

(8) 存在一条从 x_0 到 y 的可增广路 P , 令 $M \leftarrow M \oplus E(P)$, 转第(3)步;

(9) 按下式计算 a 值: $a = \min_{\substack{x_i \in A \\ y_j \in N_{G_1}(A)}} \{l(x_i) + l(y_j) - ij\}$, 修

改标号:

$$l(v) = \begin{cases} l(v) - a, & \text{若 } v \in A \\ l(v) + a, & \text{若 } v \in B \\ l(v), & \text{其它} \end{cases}$$

根据 l 求 E_l 及 G_l ;

(10) $l \leftarrow l$, $G_l \leftarrow G_l$, 转第(6)步。

Kuhn-Munkres 算法的时间复杂度是 $O(p^4)$, $p = n + m$. 求出 $G = \{X, Y, E\}$ 的最优匹配 M 后, 把 M 每条边 e_{ij} 的权值 ij 相加, 可以求得 $G = \{X, Y, E\}$ 的最大权。本文定义两个镜头 X 和 Y 的视觉相似度 $\text{Similarity}(X, Y) = \frac{\text{最大权}}{\min(n, m)}$. 使用 $\min(n, m)$ 将 $\text{Similarity}(X, Y)$ 归一化到 0, 1 之间, 值越大, 表明镜头 X 和 Y 越相似。

2.2 两个改进算法

上述 Kuhn-Munkres 算法从理论上很好地解决了两个镜头的相似度度量问题, 但在实际中, 一些镜头经常包含成百上千帧图像, 而用 Kuhn-Munkres 算法计算上百个结点构成二分图的最大权是比较耗时的工作. 考虑到镜头内部内容的冗余性, 这种计算常常也是一种不必要的操作. 因此, 必须降低镜头的帧数.

本文使用了如下两种改进算法:

(1) 子镜头构造带权二分图的算法. 把一个镜头内的内容变化分解为几个内容一致的子单元, 称为子镜头. 如静止子镜头, 可能持续上千帧, 但实际上, 仅仅使用一帧来表示它就可以了. 这里, 本文使用了文献[5]根据相机运动检测子镜头的办法, 选择或构造关键帧以降低镜头的帧数. 具体方法如表 1 所示:

表 1 基于相机运动检测的关键帧表示

Motion Type	Action
Static	Select one frame
Pan or tilt	Form a new panoramic image
Zoom	Select first and last frames
Multiple motion	Reconstruct background
Indeterministic	Select one frame

利用表 1 所示的关键帧表示办法, 可以在全面衡量两个镜头内容的前提下, 有效降低二分图的结点数, 加快 Kuhn-

Munkres 算法的计算速度. 但存在的问题是: 很多镜头仅仅包含一个子镜头和一个关键帧(如静止子镜头的一个关键帧, 平移子镜头的一个全景图等), 此时 Kuhn-Munkres 算法的使用, 就转化为求与该关键帧相似值的最大值. 因此, 可能出现颜色相似而内容不一致的矛盾, 为了解决这个矛盾, 本文使用了如下方法(2).

(2) 等间隔采样构造带权二分图的算法. 这样, 两个镜头的关键帧数目是一样的, 可以构成多结点的完全二分图. 此时使用 Kuhn-Munkres 算法, 可以达到本文的目的: 在一一对应的的前提下, 度量两个镜头内部的相似性, 从而避免了方法(1)的问题. 这种方法有可能出现关键帧的冗余表示, 如果能满足较快的检索速度, 这种冗余并不影响镜头相似度的度量.

第 3 部分的实验结果表明, 上述两种改进算法都取得了比文献[5]和常用的单关键帧表示法更好的效果, 同时检索速度也较快, 证实了二分图的最优匹配在镜头相似度度量中的有效性.

3 实验结果

镜头检索的实验数据是从电视录制的 2002 年亚运会节目, 总共有 41 分钟, 777 个镜头, 62132 帧图像. 它包含多种体育项目, 如各种球类运动、举重、游泳以及插播的广告节目等. 本文选了 7 个语义类作为查询镜头, 它们是举重、排球、游泳、柔道、划船、体操、足球, 如图 2 所示.



图 2 镜头检索的 7 个语义类例子

为了验证本文算法的有效性, 我们测试了如下 4 种方法做实验对比:

- (1) 等间隔采样构造带权二分图的算法: 取每个镜头的首帧、中间帧和末帧;
- (2) 子镜头构造带权二分图的算法;
- (3) 文献[5]的镜头检索算法;
- (4) 使用每个镜头的首帧做关键帧的镜头检索算法.

上述 4 种方法, 都使用了 HSV 的 162 个分量做颜色特征值, 使用直方图的交来度量两幅图像的相似性, 因此, 最后的实验结果能够从镜头相似度的度量方法上证明本文算法的优越性. 图 3 给出了实验程序的用户界面, 上面一行是查询视频的浏览区域, 显示视频中每个镜头的第 1 个关键帧, 用来表示每个镜头, 用户可以从中选择想要进行查询的镜头进行检索, 下面是查询结果区域. 图 3 是选择上面一行的第一个镜头(022430. bmp 游泳镜头), 按照上述方法(1)等间隔采样构造带权二分图的算法计算出的相似度最大权, 从大到小排列查询结果. 排列的顺序是从左到右, 从上到下.

实验采用了两种在 MPEG7 标准化活动中的评价指标:平均归一化调整后的检索秩 ANMRR (Average Normalized Modified Retrieval Rank) 和平均查全率 AR (Average Recall). AR 类似于传统的查全率 (Recall), 而 ANMRR 与传统的查准率 (Precision) 相比, 不仅能够反映出正确的检索结果比例, 而且能够反映出正确结果的排列序号 (详情见本文附录). ANMRR 值越小, 意味着检索得到的正确镜头的排名越靠前; AR 值越大, 意味着在前 K (K 是



图3 镜头检索实验的用户界面

也会带来检索时间的增加。

表2 本文两种方法与现有两种方法的实验对比结果

查询镜头	方法(1)		方法(2)		方法(3)		方法(4)	
	AR	ANMRR	AR	ANMRR	AR	ANMRR	AR	ANMRR
1. 举重	0.9412	0.1441	0.9412	0.1059	0.9412	0.1343	0.8824	0.1882
2. 排球	0.8556	0.2356	0.7333	0.3775	0.7333	0.3775	0.7889	0.2873
3. 游泳	0.9400	0.1597	0.9600	0.1520	0.9400	0.1522	0.9200	0.2223
4. 柔道	0.7750	0.3157	0.7750	0.3731	0.7250	0.3963	0.8500	0.2746
5. 划船	0.8125	0.2102	0.8125	0.2566	0.8125	0.2677	0.7500	0.2976
6. 体操	0.8214	0.1871	0.7857	0.1929	0.8214	0.2049	0.7500	0.2788
7. 足球	0.8421	0.2503	0.8421	0.2503	0.8421	0.2293	0.7632	0.2783
平均值	0.8554	0.2147	0.8357	0.2440	0.8308	0.2517	0.8149	0.2610

检索结果的截断值) 个查询结果中相似镜头占有所有相似镜头的比例越大. 表2是上述4种方法对7个语义镜头类的AR和ANMRR比较.

从表2可以看出, 本文的两种方法, 无论是AR, 还是ANMRR, 都取得了比现有的两种方法更好的效果, 证实了本文利用二分图的最优匹配度量镜头相似度的有效性. 方法(2)相机运动检测子镜头的结果, 使得很多镜头仅仅包括一个关键帧, 因此, Kuhn-Munkres算法的使用, 就转化为求与该关键帧相似值的最大值. 而方法(1)是对每个镜头等间隔地取3帧来刻画镜头的内容, 从而避免了方法(2)的问题, 因此方法(1)的AR和ANMRR都优于方法(2). 另外, 在PIII双CPU 1G, 256M内存的PC机上, 方法(1)的平均检索时间为15.4秒, 方法(2)的平均检索时间为9.6秒, 对于777个镜头的视频库来说, 这个检索速度还是比较快的. 当然, 也可以在一个镜头内等间隔提取更多的关键帧, 这样能够更好地刻画两个镜头的相似程度, 但这同时

4 结论

本文提出了一种基于二分图最优匹配的镜头检索方法. 我们的工作主要在于: (1) 将二分图的最优匹配用于镜头检索. 把两个镜头的相似度度量建模为一个带权的二分图: 镜头中的每一帧看成二分图的一个结点, 两个镜头之间任意帧的相似值作为边的权值, 利用最优匹配的Kuhn-Munkres算法求出该二分图的最大权, 以此作为两个镜头的相似度. 该方法的优点是在一一对应的前提下, 可以客观全面地度量两个镜头的相似度.

(2) 考虑到检索速度问题, 提出了两个改进算法. 实验对比结果证实了本文所提方法在镜头检索中的优异表现.

为了从方法上证明二分图最优匹配的有效性, 在目前的实现中, 本文仅仅使用了HSV颜色特征, 如果提取更多的特征: 如纹理、形状等, 可以取得更好的效果. 我们的下一步工作, 主要是提取更多有用的特征, 进一步提高检索的效果.

附录^[6]

平均归一化调整后的检索秩 ANMRR (Average Normalized Modified Retrieval Rank) 和平均查全率 AR (Average Recall) 的计算细节如下. 首先挑选出一个查询镜头的集合 Q , 对每个查询镜头主观地选取一组视觉相似的镜头作为标准, 正确答案 (Ground Truth). 设查询镜头 q 的相似镜头的个数为 $ng(q)$. 对于查询镜头 q , 检索结果的截断值 K 定义为 $\min(4 \times ng(q), 2 \times GIM)$, 其中, GIM 是在所有查询镜头中最大的相似镜头个

数,即 $GIM = \max\{ng(q)\}$. 对于查询镜头 q , 在前 K 个检索结果中正确的个数记为 $nr(q)$, 漏掉的个数记为 $M(q) = ng(q) - nr(q)$. 查全率记为 $R(q) = nr(q)/ng(q)$. 每个正确答案在检索结果中都有一个秩(rank) $r(i)$, $i = 1, \dots, ng(q)$. 在前 K 个检索结果中, 正确的镜头的秩 $r(i)$ 就是它的序号, 其余被漏掉的镜头的秩 $r(i)$ 都设定为 $K+1$. 对于某个查询镜头 q , 它的平均检索秩和调整后的检索秩分别定义为

$$ARR(q) = \frac{1}{ng(q)} \sum_{i=1}^{ng(q)} r(i), MRR(q) = ARR(q) - \frac{ng(q)}{2} - 0.5$$

将 $MRR(q)$ 归一化至 $[0, 1]$ 范围内, 得到归一化调整后的检索秩 $NMRR(q)$:

$$NMRR(q) = \frac{MRR(q)}{K - \frac{ng(q)}{2} + 0.5}$$

对 Q 中所有的查询镜头 q 的 $NMRR(q)$ 和 $R(q)$ 作平均, 得到 $ANMRR$ 和 AR :

$$ANMRR = \frac{1}{Q} \sum_{q=1}^Q NMRR(q), AR = \frac{1}{Q} \sum_{q=1}^Q R(q)$$

参考文献:

- [1] Ngo C W, Pong T C, Chin R T. Video partitioning by temporal slice coherency[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2001, 11(8): 941 - 953.
- [2] Liu X M, Zhuang Y T, Pan Y H. A new approach to retrieve video by example video clip[A]. Proceedings of ACM Multimedia[C]. Orlando: ACM, 1999. 41 - 44.
- [3] Wu Y, Zhuang Y T, Pan Y H. Content-based video similarity model [EB/OL]. <http://www.acm.org/sigs/sigmm/MM2000/ep/wu/wu.pdf>.
- [4] Zhao L, Qi W, Li S Z, et al. Key-Frame extraction and shot retrieval using nearest feature line (NFL) [A]. Proceedings of the International Workshop on Multimedia Information Retrieval, in Conjunction with ACM Multimedia Conference 2000[C]. Los Angeles: ACM Multimedia conference 2000.
- [5] Ngo C W, Pong T C, Zhang H J. Motion-based video representation for scene change detection[J]. Internal Journal of Computer Vision, 2002, 50(2): 127 - 143.
- [6] 林通, 张宏江, 封举富, 等. 镜头内容分析及其在视频检索中的应用[J]. 软件学报, 2002, 13(8): 1577 - 1585.
- [7] Chen L P, Chua T S. A match and tiling approach to content-based video retrieval [A]. Proceedings of IEEE International Conference on Multimedia and Expo[C]. Tokyo: IEEE, 2001. 417 - 420.
- [8] 肖位枢. 图论及其算法[M]. 北京: 航空工业出版社, 1993. 134 - 142.

作者简介:



彭宇新 男, 1974 年出生于贵州都匀, 2003 年 6 月获北京大学计算机系理学博士学位, 随后留校工作, 主要研究方向为基于内容的视频检索, 数字图像处理.

Email: peng_yuxin@icst.pku.edu.cn.

Ngo Chong Wah (杨宗桦) 男, 1971 年出生于马来西亚, 祖籍福建诏安, 2000 年 8 月获香港科技大学计算机科学系博士学位, 随后到美国伊利诺伊斯大学做博士后研究, 现为香港城市大学助理教授, 研究方向为视频和图像索引、多媒体信息检索、模式识别、计算机视觉.