

# 针对视频序列分类的快速抠像技术<sup>①</sup>

相方莉

(浙江长征职业技术学院 计算机系, 杭州 310023)

**摘要:** 针对视频序列分类的问题提出了一种快速抠像技术. 根据视频序列间的相关度进行关键帧的区分, 得到关键帧、序列间变化细微的非关键帧、序列间变化较大的非关键帧; 对于关键帧, 采用闭合式的抠像方法来进行处理, 获得透明度值、前景像素值和背景像素值; 对于变化细微的非关键帧, 提出了一种基于帧间连续性的透明度值估计和优化方法; 对于变化较大的非关键帧, 提出了一种基于特征流传递的机制来传递关键帧的有效信息. 实验结果表明, 最终在获得与传统方法相比可接受的抠像效果条件下, 这种快速抠像技术缩短了处理时间.

**关键词:** 图像序列抠像; 序列分类; 序列连贯性; 特征流

## Rapid Image Matting Method Based on Categorized Video Sequences

XIANG Fang-Li

(Department of Computer Science, Zhengjiang Changzheng Technical and Vocational College, Hangzhou 310023, China)

**Abstract:** This paper proposes a rapid image matting method based on categorized video sequences. The image sequences in a video are classified into three categories: key frames, trivial-changed frames and variant-changed frames. Different image matting algorithms are adopted to treat each image sequences category. First, the method uses closed-from image matting algorithm to process the key frames and get the alphas, foreground colors and background colors. Then, the proposed method uses the known values of key frames to estimate and optimize the alphas of the trivial-frames based on frame-coherence. After that, the matting algorithm is processed by utilizing a SIFT Flow based information propagation strategy to transfer informs from key frames to variant-changed frames. The proposed method achieves acceptable matting results and more higher efficiency compares to traditional methods.

**Key words:** image sequence matting; sequence classification; coherent sequence; SIFT flow

与图像抠像研究类似, 视频抠像受到人们的广泛关注. 在自然背景的条件下, 视频抠像是从一段视频序列中提取出动态前景元素的掩膜的过程. 为了分割的简易性, 典型的掩膜是利用了特殊的摄影棚和蓝(或绿)屏技术来实现的. 因此, 前景图像的掩膜值的提取过程便是简单地移除蓝(或绿)色像素. 尽管此方法降低了抠像问题的难度, 但这种对拍摄条件苛刻的抠像方法, 使得拍摄成本增加、难度也增加, 因此限制了其适用范围. 于是, 焦点转向了复杂背景下的自然抠像算法的研究.

Porter 等人对图像抠像提出了数学化的方程定义, 即抠像方程, 如式 1, 视频抠像对图像抠像进行了延伸.

图片抠像问题简要概括为对给定图片中每个像素估计进行透明度值的估计的过程. 每一个给定像素的颜色值都是相应的前景颜色、背景颜色以及该像素相对应的透明度值的一个线性方程组合. 对于一个三通道的 RGB 图像, 每一个像素点都能有 3 个方程和 7 个未知数. 我们需要解决的问题便是求解出图像的每一像素的透明度值、前景色值和背景色值. 在未知数远多于方程数的情形下, 抠像问题是一类欠约束问题.

$$I_p = \alpha_p F_p + (1 - \alpha_p) B_p \quad (1)$$

大部分现有的图片抠像方法, 对于这一欠约束问题的解决方案是需要用户提供附加的约束条件, 例如三分图或者草绘图(笔刷图). 这些约束条件, 提供指定

<sup>①</sup> 收稿时间:2015-03-13;收到修改稿时间:2015-05-28

了相应的确定的背景像素点和前景像素点. Wang<sup>[1]</sup>等人的抠像综述文章中大致介绍了四类图片抠像方法: 基于采样的方法、基于相干性的方法、基于采样和相干性结合的方法、基于额外信息的方法. 基于采样的方法在已知为前景和背景像素点周围采集样本点, 然后利用这些样本点来对未知像素点的透明度值进行估计. 基于相干性的方法假定前景和背景颜色都是局部平滑的, 通过对前面所述的抠像方程的优化过程来消去前景色和背景色, 从而得到掩膜值. 对上述两种方法进行结合, 利用了他们各自的优点, 能达到准确性和鲁棒性的更好的权衡, 获得更好的结果. 利用额外信息抠像的方法采用了附加的信息或者约束条件来实现抠像算法, 例如 flash matting<sup>[2]</sup>、defocus matting<sup>[3]</sup>等.

本方法的贡献: ①提出了一种针对视频序列分类的快速抠像技术; ②对分类后的序列中帧间变化细微的序列, 提出了一种快速的透明度值的估计和优化方法; ③对分类得到的序列中帧间变化较大的序列, 利用特征流的传递机制, 快速传递了关键帧的信息.

### 1 视频序列分类的抠像算法框架

算法首先对输入的视频序列进行分类, 通过计算序列之间的相关度来进行区分关键帧与非关键帧; 然后, 对关键帧应用闭合式抠像的方法, 获得关键帧的透明度值, 前景色  $F$  以及背景色  $B$ ; 接着, 处理非关键帧序列, 按照序列间变化的情形, 分为两类: 变化细微的序列、变化较大的序列, 对前一种情况, 采用基于帧间连续性的透明度值估计和优化方法来处理, 对后一种情况, 我们通过基于特征流的快速传递机制来传递关键帧的信息. 算法流程如下图所示.



图 1 算法流程图

#### 1.1 关键帧筛选

对相邻序列间相关度的计算, 如下公式所示,  $S^t$  为相邻序列的相似度;  $MIH^{t,t+1}$  表示相邻序列间的互信息量, 简化计算, 降低序列的灰度级, 256 色降为 32 色;  $D^{t,t+1}$  表示相邻序列的 HSV 颜色直方图变化差值, 充当前后帧相关度的权值, 用以降低误差.

$$S^t = MIH^{t,t+1} * (1 - D^{t,t+1}) \tag{2}$$

通过对序列的相关度进行区分, 我们可以筛选出关键帧、非关键帧中序列间变化细微的情况以及变化较大的情况. 然后分别对各种情形应用相应的方法来处理.

#### 1.2 关键帧处理

关键帧序列, 采用闭合式的抠像方法, 该方法通过输入彩色图和三分图进行抠像处理. 三分图指将图像划分为三部分区域: 确定的前景区域  $R_F$ 、确定的背景区域  $R_B$  以及二者之间的未知区域  $R_U$ . 首先选定一个深度阈值, 对深度图进行分割获得一个二值化分割结果, 小于阈值的区域作为前景; 对二值化的前景区域进行形态学腐蚀操作, 腐蚀后所获区域作为确定的前景区域  $R_F$ ; 再次对二值化前景区域进行形态学膨胀操作并求反, 所得区域作为确定的背景区域  $R_B$ ; 确定的前景、背景区域之间即为未知区域  $R_U$ , 计算其面积  $S_U$ . 计算前后连续帧之间的未知区域重叠的面积比  $P_U = (S_U^t \cap S_U^{t-1}) / \text{Max}(S_U^t, S_U^{t-1})$ ,  $P_U < 0.8$  的帧为关键帧. 输入的彩色图为原序列, 三分图的生成方法有人工标注或根据深度信息等附加信息自动获得. 闭合式抠像法<sup>[4]</sup>, 将抠像问题通过代数转化, 消去抠像方程中的前景与背景未知数, 最终通过求解一个稀疏线性方程组来求出透明度值. 抠像的过程即求解每帧未知区域  $R_U$  内的像素前景、背景和透明度  $(F, B, \alpha)$  值. 对每一视频分段, 首先用经典但较为耗时的抠像方法求解首尾两帧未知区域的  $(F, B, \alpha)$ , 然后使用快速估算方法以首尾帧为起始向前(后)推算中间普通帧的  $(F, B, \alpha)$  值, 直到在中间的某一普通帧会合. 利用该方法对关键帧进行处理, 最终可获得关键帧的透明度值, 前景色  $F$  以及背景色  $B$ .

#### 1.3 非关键帧处理

##### 1.3.1 序列间变化细微的情形

对于非关键帧中帧间变化细微的序列进行感兴趣目标的提取工作. 首先, 在预处理过程中, 会针对序列采用对应的方法获得其三分图, 用以区分序列的已知

前景、背景区域和未知区域. 我们针对序列之间变化细微的特性, 提出了以下方法进行透明度值估计的优化.

1) 透明度值估计

针对前后帧之间像素位移微小, 颜色变化不大, 考虑前后视频序列时间上的连贯性, 提出一种估计当前帧未知区域透明度值的方法. 作为一种连续性的  $\alpha$  值估计法, 记当前帧未知区域的某像素点为  $i(x,y)$ ,  $(x,y)$  为其图像中的坐标, 我们利用前一帧的相同未知的邻域透明度均值来估计此像素的透明度值, 如下式所示:

$$\alpha_{i(x,y)}^t = \frac{1}{N_\alpha} \sum_{p \in W_{N_\alpha}} \alpha_p^{t-1} \quad (3)$$

式 3 中,  $\alpha_{i(x,y)}^t$  为当前帧的像素  $i(x,y)$  的透明度值; 像素  $i(x,y)^{t-1}$  为前一帧中位于  $(x,y)$  的像素点;  $W_{N_\alpha}$  为以  $i(x,y)^{t-1}$  为中心的邻域区域内能作为有效计算候选像素的集合;  $N_\alpha$  为邻域内作为有效计算像素点的数目;  $W_{N_\alpha}$  中像素点的透明度值满足  $0 < \alpha < 1$ , 若邻域内像素点的透明度值不满足该条件, 则不属于  $W_{N_\alpha}$ .

估计出透明度值的同时, 应用类似的方法将前景像素值  $F$  和背景像素值  $B$  都估计出来, 由于关键帧的处理过程, 我们得到了相关的信息, 于是可以通过下式, 进行处理.

$$\begin{cases} F_{i(x,y)}^t = \frac{1}{N_F} \sum_{p \in W_{N_F}} F_p^{t-1} \\ B_{i(x,y)}^t = \frac{1}{N_B} \sum_{p \in W_{N_B}} B_p^{t-1} \end{cases} \quad (4)$$

如图 2, 可以表述估计方法的过程. 从左至右为视频序列中第  $t-1$  帧与第  $t$  帧的某部分, 因前后帧变化细微, 可以近似地认为前后帧之间有较大部分的相同区域, 利用前一帧中的信息来估计当前帧的信息. 因预处理过程获得了三分图, 我们只需对未知区域的像素进行估计. 用  $t-1$  帧中未知区域的某像素点的邻域信息来估计  $t$  帧中相同位置的像素点的信息.

2) 优化过程

在前一过程中, 估计出了当前帧的透明度值、前景颜色值、背景颜色值等信息. 但仅用此种方法, 获得的效果显然不够合适, 这里, 我们采用了模拟退火算法的思想, 来对透明度值进行优化. 利用前后帧的连贯性, 假定前后帧之间的背景像素值变化不大, 当前帧的背景像素值就使用估计所得的值, 不再进行优化.

于是, 在抠像方程中, 我们仅剩下两个未知数, 透明度值和前景颜色值, 我们采用模拟退火算法的思想来进行优化, 获得最终的优化值. 具体算法流程如下图所示.

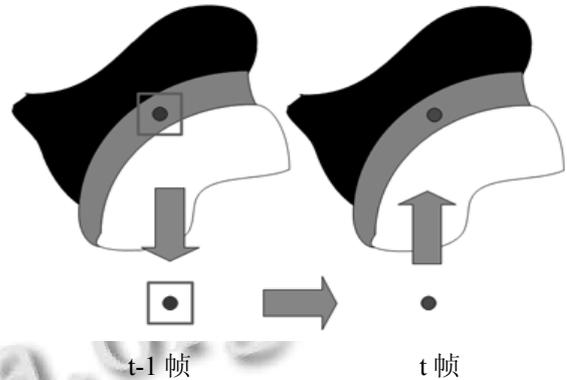


图 2 基于帧间连续的关键帧信息估计示意图

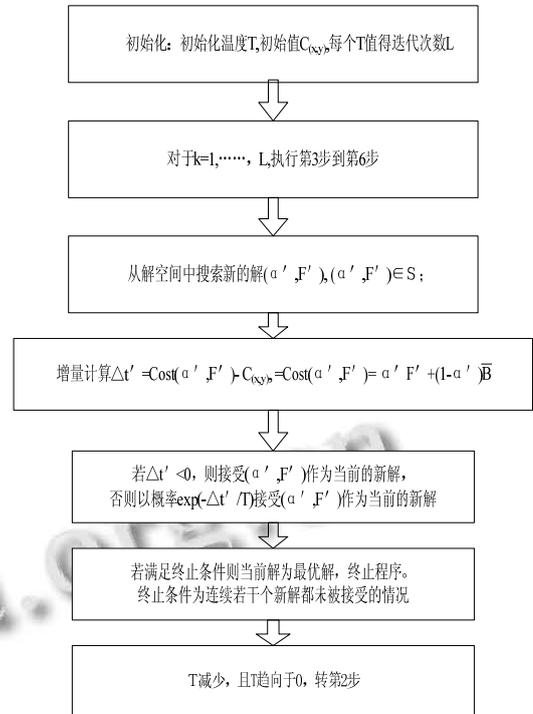


图 3 优化算法流程图

上述优化方法中, 新解产生的解系由下述方程来表述:

$$\begin{aligned} (\alpha^*, F^*) &= \arg \min_{(\alpha^*, F^*) \in S} \|\alpha^* F^* + (1 - \alpha^*) B_{i(x,y)}^t - C_{i(x,y)}^t\|, \\ S &= \{(\alpha^*, F^*) \mid \alpha^* = \bar{\alpha} \pm i\sigma_\alpha, F^* = \bar{F} \pm j\sigma_F; i, j = 0, \pm 1, \pm 2, \pm 3\} \end{aligned} \quad (5)$$

新解系  $S$  中,  $\bar{\alpha} = \bar{\alpha}_{i(x,y)}^t$ ,  $\bar{F} = \bar{F}_{i(x,y)}^t$  为  $i)$  方法中邻域窗  $W_{N_\alpha}$  和  $W_{N_F}$  内求取的均值, 同时求得的方差为  $\sigma_\alpha$ ,  $\sigma_F$ . 于是, 最优解组合可以从解系  $S$  中获得, 使得式 5 中的函数获得最小值而得到的最优解组合

$\alpha^*$ 、 $F^*$  获得的最优解以及背景像素值则组成了利用本方法处理序列间变化细微的序列的结果。

### 1.3.2 序列间变化较大的情形

对于非关键帧中序列间变化较大的情形，我们这里采用了一种基于特征流的快速传递机制，来传递关键帧的信息。这里的特征流采用了 Liu 等<sup>[5]</sup>提出的 SIFT Flow，通过利用该特征流，将关键帧信息通过前后双向传递优化，从而获得最终的结果。

#### a) 计算序列的特征流

求取前后帧序列图像的 128 维描述向量，然后进行特征匹配，求取 SIFT Flow，即特征点的流动信息。

#### b) 基于特征流的双向传递机制

此处，给定的序列  $I = \{I_1, I_2, \dots, I_{N+M}\}$  中，共有  $N+M$  帧，包含了  $N$  帧关键帧与  $M$  帧非关键帧。 $F = \{F_1, F_2, \dots, F_N\}$  表示关键帧序列， $UF = \{UF_1, UF_2, \dots, UF_M\}$ 。

前向传播，从关键帧  $F_k$  到  $F_{k+1}$  的信息流传播。依次计算前后序列间的特征流，然后根据流动的信息，将关键帧所预处理得到的结果，如前景颜色值、背景颜色值和透明度值等信息传递到后续序列中。

后向传播，从关键帧  $F_{k+1}$  到  $F_k$  的信息流传递。按照倒序的顺序求取序列之间的特征流信息，然后将关键帧预处理的信息传递后继序列当中。

然后处理并优化前后向传递过程中的误传递和信息丢失情况。将前向与后向传递的序列结果，进行相与的操作比较，从而剔除掉误传递对传递结果所造成的影响。相与操作之后，对仍然存在信息丢失的部分区域，采用种子填充技术，将该处信息补充完整。利用四联通或八联通邻域来进行填充。

最终，对这种序列间变化较大的序列子集，通过这种传递机制，获得透明度值和前景颜色值。

## 2 实验部分

### 2.1 实验环境参数设置

实验的 PC 环境 Intel Core2 Duo CPU E7500 主频 2.93GHz, 3.21GB 内存。采用 C++ 与 Matlab 实现。

实验中处理的对象为从已存在视频序列中截取的图像序列，尺寸大小为 480\*640。在变化细微的序列间的优化处理过程中，取初温  $T$  为 100，每个温度下的迭代次数  $L$  为 10，温度  $T$  的减小方法采用的是  $T=T*0.9$ 。

### 2.2 实验结果

共处理了两组视频序列，一组为利用 Kinect 采集的视频序列，能够自动的生成三分图；另一组为已有的视频序列，然后通过处理，人工标注三分图。

其中，17s 的视频序列为利用 Kinect 采集的视频序列，帧率为 23 帧/秒，共 391 幅图像，关键帧序列 20 帧，变化细微的序列子集 213 帧，变化较大的序列子集 168 帧，分别耗时 758.9s, 234.3s, 3057.6s，共耗时 4050.8s。16s 的视频序列为已有的视频序列，帧率为 23 帧/秒，共 368 幅图像，关键帧序列 57 帧，变化细微的序列子集 113 帧，变化较大的序列子集 198 帧，分别耗时 2162.9s, 124.3s, 3603.6s，共耗时 5089.8s。

下表为利用 Kinect 采集的视频序列中，处理一帧中的，变化细微的序列处理时间与两种方法单独处理的耗时对比。

表 1 序列间变化细微的情况下求取 alpha 值的耗时对比

	基于贝叶斯抠图 <sup>[6]</sup>	Shared Matting <sup>[7]</sup>	本方法
Alpha 值结果			
消耗的时间	13.169s	2.028s	1.109s

下面为处理序列间变化较大的情形，处理的视频序列为已有的视频序列。通过特征流的双向传递、优化最终得到相应结果。

对处理单帧图像，应用闭合式抠像方法，需要人工标注的先验约束，获得结果需要耗时 38s，且计算过程中需要利用到较大的计算资源来处理拉普拉斯抠像矩阵。应用这里提出的特征流传递方法，获得与前者类似的效果，耗时为 18s。



(a) 原视频序列

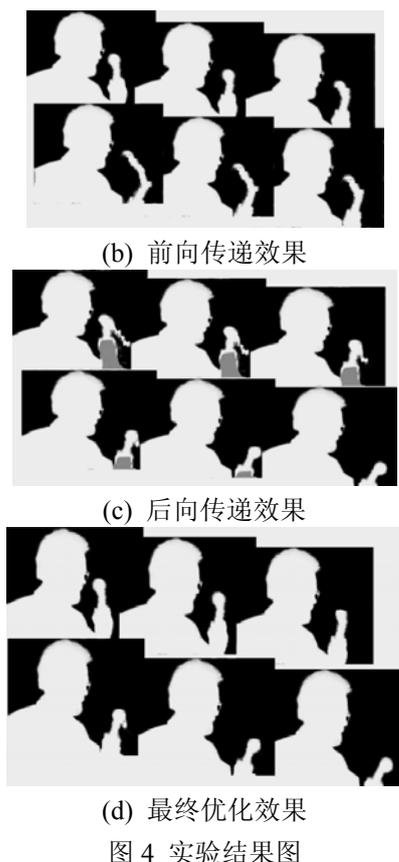
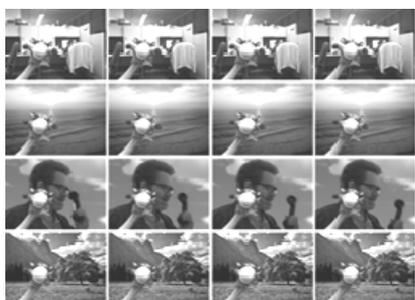


图 4 实验结果图

### 2.3 合成效果



(a) Kinect 采集视频原序列第 56-59 帧及其合成结果图



(b) 视频序列第 214-217 帧及合成结果图

图 5 合成效果图

上图为替换背景后的合成效果图。

### 3 结语

本文主要实现了对视频图像序列的感兴趣目标前景进行抠取的工作。通过对图像序列进行分类处理,将输入的图像序列分为三种情况,帧间变化较为跳跃的情形归为关键帧对象进行处理;帧间变化细微的情形,采用提出的一种基于序列间连贯性的透明度值估计和优化方法来处理;帧间变化较大的情形,利用一种基于图像序列特征流的信息传递机制来进行处理。

本文的方法能够实现与一般方法相似的、在视觉接受范围内的相近效果,但所耗费的时间较少。

### 参考文献

- 1 Wang J, Cohen M F. Image and video matting: a survey. Now Publishers Inc, 2008.
- 2 Sun J, Li Y, Kang SB, et al. Flash matting. ACM Trans. on Graphics (TOG), 2006, 25(3): 772-778.
- 3 McGuire M, Matusik W, Pfister H, et al. Defocus video matting. ACM Trans. on Graphics (TOG), 2005, 24(3): 567-576.
- 4 Levin A, Lischinski D, Weiss Y. A closed-form solution to natural image matting. IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI), 2008, 30(2): 228-242.
- 5 Liu C, Yuen J, Torralba A. Sift flow: Dense correspondence across scenes and its applications. IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI), 2011, 33(5): 978-994.
- 6 Chuang YY, Agrarwala A, Curless B, Salesin DH, Szeliski R. Video matting of complex scenes. ACM TOG, July 2002. 243-248.
- 7 Gastal ESL, Oliveira MM. Shared sampling for real-time Alpha matting. Computer Graphics Forum. Blackwell Publishing Ltd, 2010, 29(2): 575-584.