中国科学 : 信息科学　2011 年　第 41 卷　第 7 期 : 863–874
www.scichina.com　info.scichina.com
《中国科学》杂志社
SCIENCE CHINA PRESS

论　文

# 一种随机的视觉显著性检测算法

黄志勇, 何发智*, 蔡贤涛, 周正钦, 刘静, 梁铭铭, 陈晓

武汉大学计算机学院, 武汉 430074
* 通信作者. E-mail: cadcg@whu.edu.cn

收稿日期: 2010–07–08; 接受日期: 2010–11–17
国家自然科学基金 (批准号: 61070078) 和中央高校基本科研业务费专项资金资助项目

**摘要**　图像处理与模式识别技术一样, 依赖于高质量的视觉显著性图 (saliency map) 才能得到较好的处理结果. 现有的视觉显著性检测技术通常只能检测得到粗糙的视觉显著性图; 这些粗糙的视觉显著性图应用于图像处理中将严重影响图像处理的最终结果. 本文提出了一种随机的基于内容的视觉显著性区域检测算法; 该算法整合多层次粗糙的视觉显著性图到结果显著性图中, 并逐步自适应地精化可信度不高的显著性值, 最终得到一个考虑了多尺度特征的精细的视觉显著性结果. 因为随机算法具有执行效率高, 占用内存少等特点; 本文的高效随机视觉显著性检测算法不需要建立额外的辅助数据结构来加速算法, 只需占用少量内存就能快速检测出精细的高质量视觉显著性结果. 并且高效随机的视觉显著性检测算法可以直接移植到 GPU 上并行执行; 大量的实验结果表明本文的算法可以得到更加精细的显著性结果, 这些精细的显著性结果应用于基于内容的图像缩放中得到了较好的处理结果.

**关键词**　显著性检测　图像缩放　随机算法

## 1　引言

近年来, 研究者们提出了大量的基于内容的图像和视频缩放技术 [1−7]. 这些技术的目标是通过改变图像或视频的宽高比率和分辨率, 使得图像或视频适应于在目标设备上显示, 并且尽量多的保存图像和视频中的重要内容而不引入人眼可见的瑕疵. 在这些基于内容的图像和视频缩放技术中, 如何快速检测出高质量的视觉显著性区域, 仍然是一个亟待解决的具有挑战性的问题.

现有的基于像素的视觉显著性区域检测算法 [8−11] 通常都是一次计算一个像素的显著性, 计算量大; 有些算法还需要建立高维向量查找树来加速执行 [8], 这将使得算法的空间复杂度也相当高. 因此很多视觉显著性区域检测算法 [8,9] 仅仅只检测得到粗糙的视觉显著性结果. Hou 等人 [10] 和 Guo 等人 [11] 的方法都是从分析图像频谱的角度计算图像中的显著性区域; Judd 等人 [12] 则是从机器学习的角度来获取图像中的显著性区域的; 与本文的方法不同, 这些方法可以准确检测出图像中较小的目标, 倾向应用于目标识别和跟踪; 而本文方法和 Goferman 等人 [8] 的方法检测的显著性图主要应用于图像处理中.

受 Barnes 等人 [13] 和 Goferman 等人 [8] 工作的启发, 本文提出了一种新的快速检测基于内容的高质量视觉显著性区域的算法. 在该算法中, 首先使用随机查找方法快速检测出图像金字塔中每一层

引用格式: 黄志勇, 何发智, 蔡贤涛, 等. 一种随机的视觉显著性检测算法. 中国科学: 信息科学, 2011, 41: 863–874

对应的粗糙的视觉显著性区域; 其次, 精化粗糙的视觉显著性区域, 去除由于随机算法所引入的噪声; 再次, 合并图像金字塔中不同层的精化了的视觉显著性区域图; 最后, 适应性地更新每一像素的显著性值, 得到一个精细的视觉显著性结果. 本文的视觉显著性区域检测方法是一种随机算法. 它不需要建立辅助的数据结构来加速视觉显著性区域的检测, 并且仅需存储输入图像和输出的视觉显著性图的必需内存和少量额外内存就可以顺利执行. 而且它还可以快速生成与输入图像尺寸大小一致的精细的视觉显著性区域结果. 本算法可以很容易地在 GPU 上实现和并行执行. 这些特点使得本文的高效视觉显著性区域检测方法可以应用于实时地生成视频序列的视觉显著性图, 提高基于内容的视频缩放技术生成的结果的质量. 因此可以显著提高依赖视觉显著性区域检测的图像编辑操作的处理结果. 图 1 表明利用精细的视觉显著性结果, 基于内容的图像缩放算法可以得到更好的输出结果.

本文的后继部分组织如下: 第 2 节介绍了与本文研究相关的一些工作: 基于内容的图像视频缩放技术 (第 2.1 小节) 和视觉显著性区域检测 (第 2.2 小节). 第 3 节描述了本文的随机视觉显著性检测算法: 随机的视觉显著性区域检测 (第 3.1 小节)、粗糙视觉显著性区域精化 (第 3.2 小节)、多层次视觉显著性区域合并 (第 3.3 小节) 和视觉显著性区域更新 (第 3.4 小节). 第 4 节展示了大量的实验结果以及与 Goferman 等人和 Itti 等人的方法生成结果的比较. 第 5 节对本文的方法进行了总结并对未来的工作进行了展望.

## 2 相关工作

近年来, 随着移动设备的流行, 将图像或视频缩放至适合移动设备显示的尺寸的需求越来越多, 同时基于内容的图像缩放技术的研究也取得了越来越多的成果. 但是, 获取精细的视觉显著性图 (saliency map) 或重要性图 (importance map) 是一些基于内容的图像或视频缩放技术 [1,4,5,14−17] 的先决条件; 只有获得了更加精细的视觉显著性图才可能通过基于内容的图像视频缩放技术得到好的图像或视频缩放结果. 图 1 表明, 利用更加精细的视觉显著性图, Scale-and-Stretch 方法 [14] 可以生成更好的图像缩放结果. 但是利用现有显著性区域检测算法获得高质量的与输入图像尺寸一致的精细视觉显著性图不但计算量巨大, 同时还需要大量的内存用于建立高维向量搜索树来加速显著性区域的检测. 因此, 本文提出了一种能够快速检测图像中精细的视觉显著性区域的随机算法来实现快速生成与输入图像尺寸相同的精细视觉显著性图. 接下来本节将对基于内容的图像视频缩放技术和视觉显著性区域检测技术做简要介绍.

### 2.1 基于内容的图像视频缩放

基于内容的图像视频缩放技术 [1,4,7,14,18] 通过改变图像或视频的宽高比率或分辨率使得图像或视频适合于在不同尺寸的目标显示设备上显示, 并且尽量多地保存图像或视频中的重要信息. 这些图像或视频缩放技术通常都具有相同的基本结构: 首先定义图像或视频的视觉显著性图, 然后通过基于内容的算子尽量多地保存图像中的视觉重要性信息, 并且避免引入视觉可见的瑕疵. 这些算法的主要不同点在于特定的基于内容的算子. 例如, 裁切算法 [2,3,6] 查找图像或视频中覆盖重要内容的单个窗口并且保留窗口中的内容, 而完全舍弃其余部分.

与在水平和垂直方向都按比例线性缩放的一致性缩放不同, 基于内容的图像视频缩放技术, 如非真实感缩放技术 (non-photorealistic retargeting)、线裁剪技术 (seam carving)、单向图像扭曲技术 (one-directional image warping) 和全向图像扭曲技术 (omnidirectional image warping) 允许非线性不

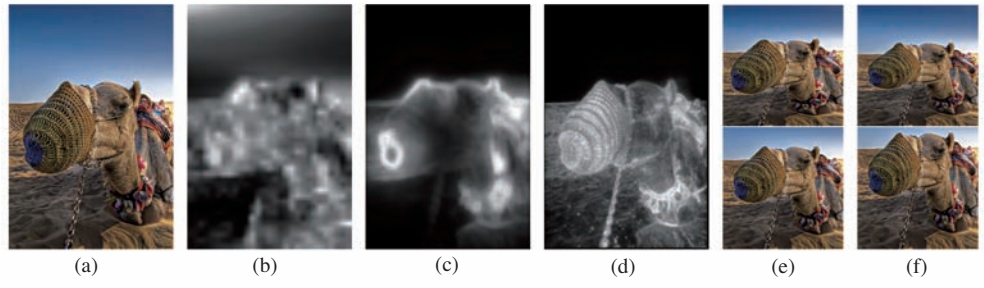


**图 1 视觉显著性区域结果比较与图像缩放结果比较**

**Figure 1** Comparisons of saliency map results and retargeting results

(a) Input image with dimensions $476 \times 704$; (b) saliency map produced using the method of Itti et al.; (c) saliency map generated from the algorithm by Goferman et al., as produced by Stas Goferman.; (d) saliency map detected using our random algorithm; (e) upper image is the result produced using the saliency map in (b), lower image is the result generated from the saliency map in (c); (f) upper image is the uniform scale result of the input image, and the lower image is the result produced using the saliency map in (d). The retargeting results of the input image were produced using the scale-and-stretch method except for the uniform scale result

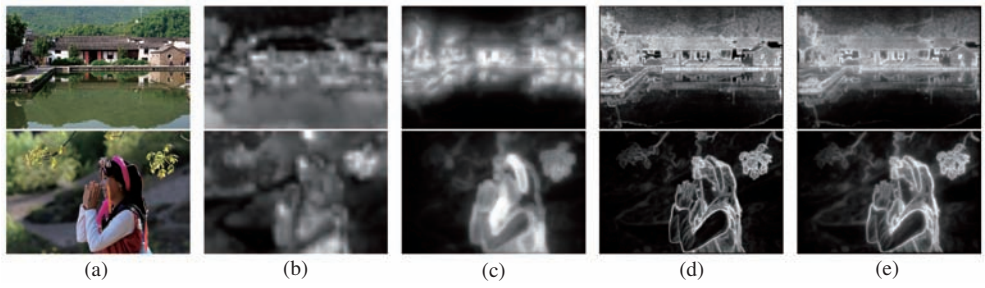


**图 2 结果显著性图比较**

**Figure 2** Comparison of saliency map results

(a) Input image with dimensions $742 \times 495$; (b) saliency map produced using the method of Itti et al.; (c) saliency map generated from the algorithm by Goferman et al., as produced by Goferman; (d) coarse saliency map detected with our random algorithm in a single layer; (e) refined saliency map obtained with our random algorithm using multiple layers of the Gaussian image pyramid. (b) and (c) were generated with a coarse scale and then uniformly scaled to size $742 \times 495$. (d) was generated with a single fine scale. (e) is the result of applying our random efficient saliency map detection method to the input image

一致地缩放图像中的不同部分. 它们都致力于根据图像中不同部分的视觉重要性属性来重新分布图像中的像素点. Wang 等人 [14] 提出的方法允许用户即使只调整图像的宽或高, 也能局部地在水平和垂直方向缩放图像中的内容. Cho 等人 [19] 研究了在基于块的相关性和完整性的约束下图像像素的重新分布方法. 这些算法为图像编辑操作 (包括图像缩放) 提供了更多的灵活性, 但是这些算法的时空复杂度也相当高. 通过改进重要性图或视觉显著性图, 在重要性信息图或视觉显著性图模型中考虑运动信息, 以及在单帧上在考虑时空相关性的基础之上对单帧进行基于内容的图像操作, 几乎所有的基于内容的图像缩放方法都可以应用到基于内容的视频缩放中.

### 2.2 图像视觉显著性检测

图像的显著性区域检测是计算机视觉中的挑战性难题之一. 大量依赖于显著性区域的应用导致很多不同的显著性区域定义和感兴趣区域的检测算法存在. 这些显著性区域检测算法关注于找出人类观察者第一眼所注意到的固定点或对象. 这类显著性对于理解人类关注点和特定的应用, 如自动聚焦等都非常重要. 其他的显著性检测算法更加注重于检测图像中的单个主要对象.

Itti 等人 [9] 根据早期原始视觉系统的行为和神经网络结构提出了一个视觉关注系统. Itti 等人的算法组合多尺度的图像特征到一个单一视觉显著性图中. 这些多尺度的图像特征包括 6 个亮度特征图、12 个彩色和 24 个方向特征图. 为了快速地检测这些多尺度的图像特征, Itti 等人仅在一些粗糙的层次上计算了这些粗糙的特征图. 实际上, Itti 等人的方法只生成了一个粗糙的视觉显著性区域图. 图 2(b) 是使用 Itti 等人提出的方法生成的一个显著性区域图. 它与本文方法不同, 本文的方法是快速生成了一个精细的与输入图像尺寸相同的显著性图.

Goferman 等人 [8] 提出了一种基于上下文的显著性, 它的目标是检测图像中代表场景的图像区域. 他们认为像素的显著性应该由以该像素为中心的图像块表示或相关, 因为它反应了像素所在位置的图像上下文信息. 这样, 如果以像素 $i$ 为中心的图像块 $p_i$ 与图像中所有其他的图像块差异都非常大, 那么像素 $i$ 被看作是显著的.

定义 $d_{\text{color}}(p_i, p_j)$ 为向量化了的图像块 $p_i$ 和 $p_j$ 之间在 CIE Lab 颜色空间中的 Euclid 距离, 并且归一化到 $[0, 1]$ 的范围内; 当 $d_{\text{color}}(p_i, p_j)$ 对于任意的相像块 $p_j$ 都非常大时, 认为像素 $i$ 是显著的.

定义 $d_{\text{position}}(p_i, p_j)$ 为图像块 $p_i$ 和 $p_j$ 所在位置之间的 Euclid 距离, 并且也被归一化到 $[0, 1]$ 的范围内. 基于上面的观察, Goferman 等人定义了一个衡量一对图像块的不相似性的方法为

$$d(p_i, p_j) = \frac{d_{\text{color}}(p_i, p_j)}{1 + c \times d_{\text{position}}(p_i, p_j)}. \tag{1}$$

在 Goferman 等人提出的方法中, 只考虑 $K$ 个最相似的的图像块 (如果最相似的图像块都明显地不同于图像块 $p_i$, 那么显然, 图像中的所有图像块都明显地不同于图像块 $p_i$). 因此, 对于每一个图像块 $p_i$, 在输入图像中根据公式 (1) 找出 $K$ 个最相似的图像块, 并根据下面的公式计算位置 $i$ 处像素的显著性:

$$S = 1 - \exp\left\{ -\frac{1}{K} \sum_{k=1}^{K} d(p_i, p_k) \right\}. \tag{2}$$

在 Goferman 等人的实现中只检测了一个宽或高 (较大者缩放至 256) 为 256 个像素的粗糙的显著性区域图; 它的计算量非常大, 而且为了加速最相似图像块的查找, 还需要在稀疏的网格上建立高维向量查找树. 因此, 如果在精细的原图上计算每一个像素的显著性值将会非常低效. 本文的思想受 Goferman 等人提出的算法的启发, 但是不是在整个图片中查找最相似的图像块来计算像素的显著性, 而是随机地从图像中提取 $2K$ 个图像块, 并且保留其中的 $K$ 个与图像块 $p_i$ 最相似的图像块, 舍弃其他图像块. 本文的方法只需要随机地收集 $2K$ 个图像块, 不需要建立辅助的数据结构来加速查找. 与 Goferman 等人的算法相比, 本文的方法执行速度快, 占用内存少.

Itti 等人的方法和 Goferman 等人的方法都能有效地检测出输入图像的视觉显著性区域; 但是因为在精细的原图上直接检测视觉显著性图计算量巨大而且还需要占用非常多的内存, 所以它们只检测了一个粗糙的视觉显著性图. 而本文的方法能够快速地在精细的原图上直接生成精细的视觉显著性图. 并且精细的视觉显著性图更适合于基于内容的图像或视频缩放操作. 图 1 中显示了使用精细的视

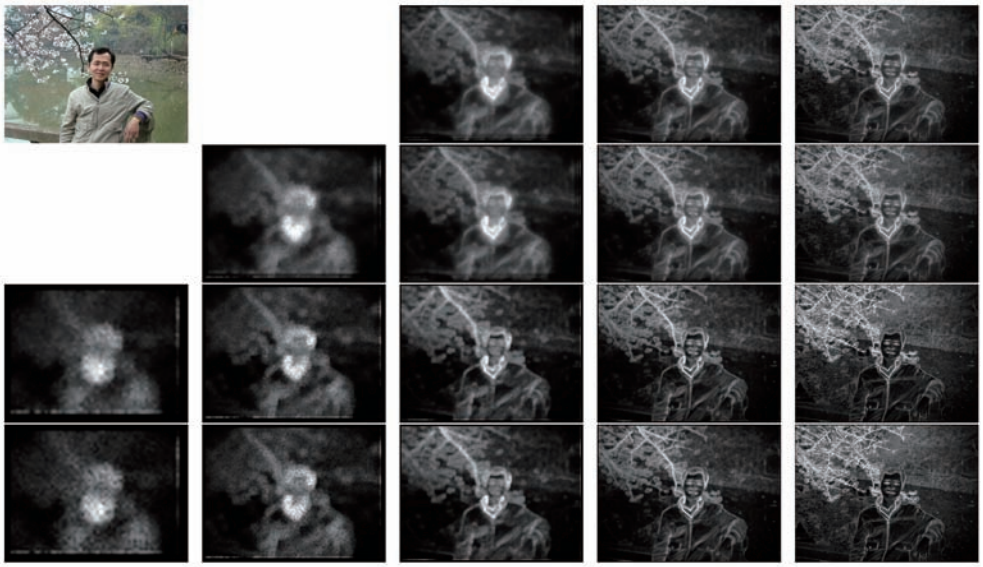


**图 3 对应于图像金字塔中各层和本文算法中各阶段的视觉显著性结果图**

**Figure 3** Multiple layer saliency map results, corresponding to the layers in a Gaussian image pyramid of the input image. In this experiment, there are 5 layers in the Gaussian image pyramid. Images in column 1 (Layer 0) correspond to the coarsest layer in the image pyramid except the color image, which is the input image. Images in column 5 (Layer 4) are the finest saliency maps corresponding to the finest layer of the image pyramid. Columns 1 (Layer 0) to column 5 (Layer 4), correspond to layers of the image pyramid from the coarsest to the finest layers, respectively. Images in the bottom row (row 4) are the direct random saliency maps for different layers of the image pyramid. They are coarse saliency maps. Images in the third row (row 3) are the refined coarse saliency maps produced from the coarse saliency maps in the bottom row. Images in the second row (row 2) are merged saliency maps of the current layer refined saliency map and the nearest coarse layer refined saliency map. Images in the upper row (row 1) are updated saliency maps, except for the color image. The updated saliency map is obtained from the merged saliency map, after updating any unreliable saliency values. The upper rightmost saliency map is the output saliency map of our method.

觉显著性图利用 Scale-and-Stretch 方法进行基于内容的图像缩放结果; 结果表明, 精细的显著性图能够生成更好的图像缩放结果. 图 2 表明, 与 Goferman 和 Itti 等人的方法相比, 本文方法可以得到更加精细的视觉显著性图.

## 3 随机的视觉显著性检测

本文的随机视觉显著性区域检测方法包括 4 个阶段. 第 1 阶段计算对应于输入图像的图像金字塔中各层的多层次的粗糙显著性图. 第 2 阶段精化各层粗糙的显著性图, 去除显著性图中由于随机算法引入的噪声. 第 3 阶段合并精化了的多层次显著性图到一个合并后的显著性图中, 得到考虑了多尺度特征的显著性图. 第 4 阶段适应性地更新可信度不高的显著性区域, 进一步增强合并后的显著性图结果. 对显著性图精细程度要求不高, 但要求较高的检测速度时, 可以只采用本文的随机显著性检测算

法的第 1 和第 2 阶段快速高效地生成粗糙的显著性图. 当需要精细的与原输入图像尺寸相同的显著性图时, 可以使用本文算法的全部 4 个阶段考虑多尺度的图像特征, 生成高质量的视觉显著性结果图.

使用随机显著性图 RSM (random-saliency map) 作为图像 2D 坐标的函数 $f : P \mapsto S$; 它定义在输入图像上所有可能的像素位置坐标上; $S$ 是归一化了的显著性值. 给定在输入图像 $P$ 中坐标为 $p$ 的图像块, 它对应的显著性值 $s$ 在 $[0,1]$ 范围内, $f(p) = s$. 函数 $f$ 的值是被归一到 $[0,1]$ 之间的显著性值, 它们被存储在一个与图像 $P$ 尺寸相同的二维数组中.

### 3.1 随机的显著性检测

令 $s = f(p)$, 通过测试一系列的与点 $p$ 的距离成指数递减的备选位置的图像块来尝试计算位置 $p$ 处的显著性:

$$p_i = p + \omega\alpha^i R_i,$$

其中 $R_i$ 是均匀分布的随机变量, 其取值范围在 $[-1,1] \times [-1,1]$, $\omega$ 是图像的宽高尺寸的一半, $\alpha$ 是一个固定的随机选择窗口尺寸的衰减因子; 测试从 $i = 1$ 到 $2K$ 的图像块, 直到当前的查找半径 $\omega R_i$ 小于 1 个像素为止. 此时, 如果 $i$ 仍然小于 $2K$, 则重置 $i = 1$, 直到测试的候选块的数量达到 $2K$ 为止. 在本文的实现中, $\alpha = 0.75$.

在这 $2K$ 个候选块中, 根据方程 (1) 计算候选块与块 $p$ 的不相似性 $d_i$. 我们只保留 $2K$ 个候选图像块中 $K$ 个不相似性值小的图像块, 舍弃其他图像块. 根据方程 (2) 用这 $K$ 个保留的图像块来计算坐标 $p$ 处的显著性值 $s$, 在本文的实验中 $K = 32$. 按与点 $p$ 的距离成指数递减的随机位置选取的 $2K$ 个候选块只是图像中所有可能的候选块中的一小部分, 尽管这 $2K$ 个候选块的选取符合与图像块 $p$ 越邻近的位置出现高相似图像块的可能性更大的特点, 但是它仍然是一种不完全采样, 所以会给检测得到的显著性图引入误差. 但随着采样数 $2K$ 增大时, 误差会显著减小. 图 3 的第 4 行显示了使用本文的随机显著性方法在图像金字塔的单层上检测得到的粗糙的视觉显著性图.

### 3.2 精化粗糙的视觉显著性图

在使用随机算法从图像金字塔的单层中检测得到的粗糙视觉显著性图中, 由于采样量不足, 存在大量由于随机算法引入的噪声. 图 3 中的最后一行图像, 是直接检测出的粗糙的显著性图, 从中我们可以看出这些粗糙的视觉显著性图中存在大量噪声. 噪声是由于使用了随机算法随机地选择 $2K$ 个图像块进行计算显著性结果而引入的. 我们尝试使用双边滤波器及中值滤波器来减少噪声. 这些滤波器都能较好地减少图像中的噪声. 但是它们的执行效率相对较低. 所以尝试使用 8 邻域的显著性值来选择性地精化粗糙的视觉显著性图. 因为随机选择的备选图像块可能与位置 $p$ 处的图像块的差异非常大, 这将使得位置 $p$ 处的显著性值比实际的显著性值大; 而过度地在与位置相邻的区域内选择候选图像块将可能导致所有取得的候选图像块都与位置 $p$ 处的图像块非常相似, 导致位置 $p$ 处的显著性值偏小; 在这样的情况下, 再经过归一化处理, 将使得得到的粗糙的显著性图中存在大量噪声. 但是相邻位置间的显著性应该相关且相近, 因此主要需要精化那些与周围邻域间显著性差异非常大的像素位置处的显著性值. 我们选择精化那些与周围 8 邻域间差异非常大的显著性值, 因为它们的可信度不高. 图 3 的第 3 行中的图像都是通过本节方法精化上节方法得到的粗糙的显著性图的结果. 图 3 表明, 精化了的视觉显著性图明显比粗糙的显著性图更加清晰平滑.

### 3.3 多层次视觉显著性区域合并

精化了的显著性区域图仍然不能完全去除噪声. 为了整合多尺度显著性特征至最终的结果显著性图中, 组合最邻近的粗糙层中的精化了的显著性图和当前层的精化了的显著性图至合并的显著性图中.

$$\mathrm{RSM}^{i}_{\mathrm{merged}} = \mathrm{merge}(\mathrm{RSM}^{i}_{\mathrm{refined}},\ \mathrm{RSM}^{i-1}_{\mathrm{refined}}),$$

其中, $\mathrm{RSM}^{i}_{\mathrm{merged}}$ 是第 $i$ 层的合并了的随机显著性图; $\mathrm{RSM}^{i}_{\mathrm{refined}}$ 是第 $i$ 层中精化了的随机显著性图. 在图 3 中, 第 2 行中的图片都是合并不同层的精化了的显著性图得到的结果, 第 3 行中的图片是不同层的精化了的显著性图. $\mathrm{RSM}^{i-1}_{\mathrm{refined}}$ 是第 $i$ 层的最邻近的精化了的显著性图. 由于 $\mathrm{RSM}^{i-1}_{\mathrm{refined}}$ 和 $\mathrm{RSM}^{i}_{\mathrm{refined}}$ 的尺寸不同, 因此在进行合并时, 需要将 $\mathrm{RSM}^{i-1}_{\mathrm{refined}}$ 的尺寸缩放到与 $\mathrm{RSM}^{i}_{\mathrm{refined}}$ 相同的尺寸.

我们尝试了使用加权的显著性图合并方法和平均的显著性图合并方法来合并两个精化了的显著性图. 实验结果表明加权的显著性图的合并方法可以得到更优的合并结果. 在第 $i$ 层的坐标 $p$ 处, 精化了的显著性值为 $S^{i}_{(r,p)}$, 其中 $S^{i}_{(r,p)}$ 被归一化到了 $[0,1]$. 为了计算在第 $i$ 层中的合并了的显著性值, 第 $i-1$ 层的粗化了的显著性图被缩放到了与第 $i$ 层中显著性图一致的尺寸. 使用 $S^{i-1}_{(r,p)}$ 表示在缩放了的显著性图 $\mathrm{RSM}^{i-1}_{\mathrm{refined}}$ 中, 位置 $p$ 处的显著性值. 这样我们可以定义相邻两层的精化了显著性值的合并结果为

$$S^{i}_{(m,p)} = w^{i}_{p} S^{i}_{(r,p)} + w^{i-1}_{p} S^{i-1}_{(r,p)}, \tag{3}$$

$$w^{i}_{p} + w^{i-1}_{p} = 1.0, \tag{4}$$

$$\frac{w^{i}_{p}}{w^{i-1}_{p}} = \frac{S^{i}_{(r,p)}}{S^{i-1}_{(r,p)}}. \tag{5}$$

根据方程 (4) 和 (5), 在 $S^{i}_{(r,p)}$ 和 $S^{i-1}_{(r,p)}$ 已知的情况下, 可以计算出 $w^{i}_{p}$, $w^{i-1}_{p}$, 当 $S^{i-1}_{(r,p)} = 0.0$ 时, $S^{i}_{(m,p)} = S^{i}_{(r,p)}$. 这样, 可以根据公式 (3) 计算出合并后的显著性值. 在图 3 中的第 2 行的图片都是合并了的显著性图. 它表明合并了的显著性图整合了两个不同层次上的多尺度显著性特征, 得到的显著性结果比精化后的显著性图更加平滑清晰.

### 3.4 视觉显著性区域更新

通常, 合并后的视觉显著性图已经足够精细, 适合用于一些基本的图像编辑算法. 但是, 某些情况下, 可能希望使用更加精细的视觉显著性图来获得更佳的图像处理效果. 此时可以使用本节的更新显著性图的方法进一步增强合并了的显著性图. 与第 3.2 小节相同, 我们认为在位置 $p$ 处, 更高的显著性值具有更高的可信度. 因此, 我们使用在最邻近的粗糙层中的合并了的显著性图来更新当前层的合并了的显著性图. 在实验中, 我们选择两个合并了的显著性图中对应位置 $p$ 处的较大的显著性值作为最终更新后的显著性值. 图 3 中的第 1 行中的图片除了第 1 张彩色图外, 其他的都是更新后的显著性图; 从图中我们可以看出更新后的显著性图更加精细.

## 4 结果与讨论

我们使用 C++ 程序设计语言, 在 MS Windows 环境下实现了本文的算法; 同时也采用 CUDA 实

现了本文算法的 GPU 版本. 并且使用本文算法根据输入图片生成了大量的显著性结果图片. 本文的所有的结果图片都是在一台 Pentium® Dual-Core CPU E5300 个人电脑上生成的, 并行代码是在一块 NVIDIA GeForce 9800GT 显卡上运行的. 本文中的大量视觉显著性检测结果表明, 本文的方法在各种不同的输入图片上都可以获得较好的视觉显著性检测结果, 而且本文的方法生成的显著性结果图片相对于 Itti 等人和 Geforman 等人的方法, 检测得到的视觉显著性图理加精细. 图 5 和 6 显示了本文方法生成的更多的显著性结果.

图 1 表明, 本文方法生成的与输入图片尺寸相同的精细的视觉显著性图应用于基于内容的图像缩放中, 可以得到比使用 Itti 和 Goferman 等人的方法生成的显著性图获得的结果图片更加好的图像缩放效果; 并且也说明使用更加精细的显著性图, 输入图像的显著性区域在缩放的结果图片中会被保存得更好.

图 2 表明使用本文的方法生成的视觉显著性图结果比使用 Goferman 等人的基于内容的方法检测出的显著性图及 Itti 等人的方法生成的显著性图更加清晰精细.

图 3 显示了本文方法在不同阶段生成的多层次的显著性图. 它表明本文的方法生成的显著性图是一种多层次多尺度的显著性图. 因为考虑多层次多尺度的特征, 本文的方法可以生成增强的精细的显著性图. 从图 3 中, 可以看出, 在精细的层次上, 本文的随机的显著性检测方法仍然可以生成精细的结果, 并且它也可以直接应用于图像编辑方法. 在图 3 中, 第 3, 4 层中的图片表明, 在图像金字塔的精细层中, 本文方法得到的显著性结果类似于图像边缘检测的结果, 这与本文方法中采用的图像块的大小相关. 当图像块半径参数越小时, 显著性结果与边缘检测结果更相似. 另外, 虽然本文中方法采用了合并多层粗糙的显著性图来获取多尺度的显著性信息, 但是在多层次的图像显著性图中, 底层 (第 0,1 层) 的显著性并没有很好的传递到上层 (第 3, 4 层) 的合并和更新了的结果显著性图中.

图 4 将本文的方法与 Itti 等人的方法及 Goferman 等人的方法作了比较. 结果表明, 处理同一张 800×600 的输入图片, 本文的方法的处理时间明显少于 Itti 等人和 Goferman 等人的方法; 而本文方法的 GPU 版本所需执时间则只需 17 ms. 处理同一张 800×600 的输入图片时, 所占用的内存比较表明, 使用 Goferman 等人的方法所占用的内存最大, 而使用本文的方法所占用的内存则比较小, 本文方法的 GPU 版本所占用的内存最小, 它所占用的内存不包括执行时其所使用的显卡内存.

图 5 显示了本文的方法生成的显著性图与 Itti 等人的方法生成的显著性图的比较, 结果表明本文的方法生成的显著性图结果更加精细. 图 6 展示了本文方法生成的更多的显著性结果.

## 5 结论与展望

本文提出了一种随机的基于内容的视觉显著性图检测算法; 该方法可以快速地生成与原输入图像尺寸相同的精细的显著性图. 与本文方法不同, Goferman 等人和 Itti 等人的方法仅仅只在一个粗糙的层次上生成了一个粗糙的显著性图.

本文的方法甚至可以快速生成可用于实时的基于内容的视频缩放的显著性图序列, 并且本文的方法也可以生成用于基于内容的图像缩放的与输入图像尺寸一致的精细的显著性图. 本文提出的快速随机显著性区域检测算法包括以下 1 个阶段: 第 1 阶段采用随机算法生成粗糙的显著性图; 第 2 阶段对生成的精糙的显著性图进行精化, 移除随机噪声; 第 3 阶段合并粗糙的显著性图以获得多尺度的结果; 最后一阶进一步增强合并的显著性图得到一个高质量的精细的显著性图.

在未来的工作中, 我们将进一步研究如何将粗糙的区域显著性从底层传递到最后的结果显著图中,

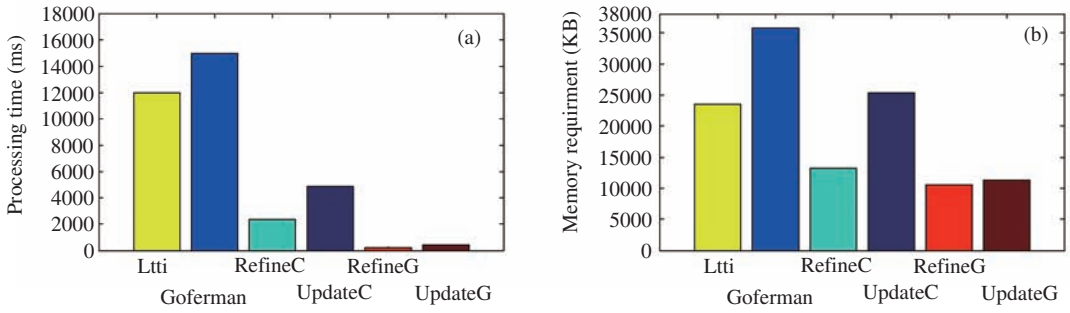


**图 4 本文的方法与 Itti 等人的方法及 Goferman 等人的方法处理同一张尺寸为 800×600 的输入图片时的处理时间与消耗内存的比较**

**Figure 4** Comparisons of processing time and memory requirements for processing an image with dimensions $800 \times 600$. (a) Processing time; (b) memory requirements


**图 5 本文方法与 Itti 等人的方法生成的显著性结果比较. 第 1 列是输入图像; 第 2 列是 Itti 等人的方法生成的显著性结果图; 第 3 列是本文的方法生成的显著性结果图**

**Figure 5** Comparison of saliency map results. Images in the first column (a) are the input images, in the second column (b) the results generated using Itti's method, and in the third column (c) the saliency maps which were produced using our method


**图 6　更多的视觉显著性图检测结果. 第 1 列和第 3 列是输入图片; 第 2 列和第 4 列是本文方法生成的显著性结果图**

**Figure 6**　Additional example. Images in the first and third columns ((a) and (c)) are input images. Images in the second and the fourth columns ((b) and (d)) are the resulting saliency maps produced using our random saliency map detection method

以进一步增强检测到的显著性结果图.

**致谢**　感谢 Stas Goferman 帮助我们生成了本文中引用的 3 幅基于上下文的显著性图; 感谢 Chinagraph2010 的评审专家, 他们为本文提出了大量宝贵的修改意见.

## 参考文献


1 Avidan S, Shamir A. Seam carving for content-aware image resizing. ACM Trans Graph, 2007, 26: 1–8
2 Chen B, Sen P. Video carving. In: Proceedings of Eurographics. Hersonisso, 2008
3 Liu H, Xie X, Ma W, et al. Automatic browsing of large pictures on mobile devices. In: Proceedings of the 11th ACM International Conference on Multimedia. Berkeley, 2003. 148–155
4 Pritch Y, Kav-Venaki E, Peleg S. Shift-map image editing. In: Proceedings of the 12th IEEE International Conference on Computer Vision. Kyoto, 2009. 151–158
5 Rubinstein M, Shamir A, Avidan S. Improved seam carving for video retargeting. ACM Trans Graph, 2008, 27: 1–9
6 Santella A, Agrawala M, DeCarlo D, et al. Gaze-based interaction for semi-automatic photo cropping. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Montréal, 2006. 771–780
7 Wolf L, Guttmann M, Cohen-Or D. Non-homogeneous content-driven video-retargeting. In: Proceedings of the 11th IEEE International Conference on Computer Vision. Rio de Janeiro, 2007. 1–6
8 Goferman S, Zelnik-Manor L, Tal A. Context-aware saliency detection. In: IEEE Conference on Computer Vision and Pattern Recognition. San Francisco, 2010. 2376–2383
9 Itti L, Koch C, Niebur E. A model of saliency-based visual attention for rapid scene analysis. IEEE Trans Patt Anal Mach Intell, 1998, 20: 1254–1259
10 Hou X, Zhang L. Saliency detection: a spectral residual approach. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR'07, 2007. 1–8
11 Guo C L, Ma Q, Zhang L M. Spatio-temporal saliency detection using phase spectrum of quaternion Fourier transform. In: IEEE Conference on Computer Vision and Pattern Recognition, 2008. 116
12 Judd T, Ehinger K, Durand F, et al. Learning to predict where humans look. In: IEEE International Conference on Computer Vision (ICCV2009). http://people.csail.mit.edu/tjudd/WherePeopleLook/index.html
13 Barnes C, Shechtman E, Finkelstein A, et al. PatchMatch: a randomized correspondence algorithm for structural image editing. ACM Trans Graph, 2009, 28: 1–11
14 Wang Y, Tai C, Sorkine O, et al. Optimized scale-and-stretch for image resizing. ACM Trans Graph 2008, 27: 118
15 Chen B, Lee K, Huang W, et al. Capturing intention-based full-frame video stabilization. In: Computer Graphics Forum 2008, vol 27. Oxford: Blackwell Science Ltd. 1805–1814
16 Liu L, Chen R, Wolf L, et al. Optimizing photo composition. In: Computer Graphics Forum, 2010, vol 29. 469–478
17 Wang Y, Lee T, Tai C. Focus+ context visualization with distortion minimization. IEEE Trans Visual Comput Graph, 2008, 14: 1731–1738
18 Wang Y, Fu H, Sorkine O, Let al. Motion-aware temporal coherence for video resizing. In: ACM SIGGRAPH Asia, 2009. 1–10
19 Cho T, Butman M, Avidan S, et al. The patch transform and its applications to image editing. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2008. 1–8


# Efficient random saliency map detection


HUANG ZhiYong, HE FaZhi*, CAI XianTao, ZOU ZhengQin, LIU Jing, LIANG MingMing & CHEN Xiao

*Computer School, Wuhan University, Wuhan* 430074*, China*
*E-mail: cadcg@whu.edu.cn



**Abstract** Most image retargeting algorithms rely heavily on valid saliency map detection to proceed. However, the inefficiency of high quality saliency map detection severely restricts the application of these image retargeting methods. In this paper, we propose a random algorithm for efficient context-aware saliency map detection. Our

method is a multiple level saliency map detection algorithm that integrates multiple level coarse saliency maps into the resulting saliency map and selectively updates unreliable regions of the saliency map to refine detection results. Because of the randomized search, our method requires very little additional memory beyond that for the input image and result map, and does not need to build auxiliary data structures to accelerate the saliency map detection. We have implemented our algorithm on a GPU and demonstrated the performance for a variety of images and video sequences, compared with state-of-the-art image processing.

**Keywords** saliency map detection, image retargeting, random algorithm