

# 显著性检测领域之图像特征

朱亚菲

2015 年 1 月

## 目录

1 引言	2
2 基于像素点的 feature contrasts	3
2.1 颜色特征	3
2.1.1 彩色模型	3
2.2 SIFT 特征	3
3 基于区域的 feature contrasts	4
3.1 颜色	4
3.1.1 Color Contrast	4
3.1.2 颜色直方图	5
3.2 纹理特征	6
3.3 位置特征	7
3.4 形状特征	7
3.4.1 区域面积	7
3.4.2 奇异值特征	8
3.4.3 HOG 特征	8
3.5 Visual Complexity Contrast	10
3.6 Background Weighted Contrast	11
4 Feature Contrasts 的融合	11
5 Priors	12
5.1 Backgroundness	12
5.2 Center Prior 或 Location Prior	12
5.3 Color Prior	12

## 1. 引言

图像特征提取是图像分析与图像识别的前提，它是将高维的图像数据进行简化表达最有效的方式，从一幅图像的  $M \times N \times 3$  的数据矩阵中，我们看不出任何信息，所以我们必须根据这些数据提取出图像中的关键信息，一些基本元件以及它们的关系。

图像特征可分为全局特征和局部特征。其最大的区别是特征提取的空间范围不同。全局特征是从整个图像中提取的特征，而局部特征是从图像区域中提取的特征。全局特征容易受到环境的干扰，光照、旋转、噪声等不利因素都会影响全局特征。相比而言，局部特征点，往往对应着图像中的一些线条交叉、明暗变化的结构中，受到的干扰也少。

总的来说，全局特征是对图像内容的高度抽象的概括。如果用户对整个图像的整体感兴趣，而不是对前景本身感兴趣的话，用全局特征来描述图像是比较合适的。但是无法分辨出前景和背景却是全局特征本身就有的劣势，特别是在关注的对象受到遮挡等影响的时候，全局特征很有可能就被破坏掉了。

在显著性检测中，由于关注的是显著目标，并且是在同一幅图像中进行中央-周围/局部/全局对比，而不是在图像间进行比较，因此用到的应该都是局部特征。显著性检测领域所说的局部和全局方法是指对某一像素或区域，在计算其对比度时是与周围相比还是与图像中所有其它像素或区域相比。

基于像素级或 block 级的显著性检测方法中用到的是 center-surround contrast 或 global contrast，而基于区域的方法用到的则是 local contrast 或 global contrast，这是因为区域形状不规则，所以叫 local contrast。

英国 Oxford 大学的 Andrea Validi，他是 VLFeat 的发起者和主要作者。VLFeat 官方主页：[vlfeat.org](http://vlfeat.org)。它实现了一个计算机视觉领域诸多算法的开源库，包括 SIFT，HOG 等等。底层代码用 C 语言实现，并提供了 MATLAB 接口。支持 Windows，Mac OS X 和 Linux。VLFeat 目前正在逐渐实现其他常用的特征描述子。和 OpenCV 相比，VLFeat 是一个轻量级的库，主要实现了在特征提取和聚类方面的高效算法，可以用在图像检索和物体识别领域中。

<http://www.sigvc.org/bbs/thread-165-1-1.html>

图像特征又可以分为低层次特征和高层次特征。其中低层次特征是不需要任何形状信息（空间关系的信息）就可以从图像中自动提取的基本特征。高层次特征提取关心的是在图像中找出形状。例如，要自动识别人脸，一种方法是提取组成部分特征。也就是说，需要提取眼睛、耳朵和鼻子这些主要的脸部特征。这些特征可以利用它们的形状找到：眼睛的白色部分是椭圆形的；嘴巴可以看做是两条直线，眉毛也一样。形状提取意味着找出它们的位置、朝向和尺寸。所有低层次方法都可以应用于高层次特征提取，从而在图像中找到形状。

## 2. 基于像素点的 feature contrasts

### 2.1 颜色特征

#### 2.1.1 彩色模型

彩色模型 (也称为彩色空间或彩色系统) 的目的是在某些标准下用通常可以接受的方式方便地对彩色加以说明。

##### 1) RGB 颜色模型

RGB 颜色空间是常用的表示彩色图像的一种颜色空间, 它是以红、绿、蓝三种颜色为基础, 亦称为“三原色”。所谓的“原色”是一种生物学概念, 是根据人眼对光线感知的生理作用来定义的。每一种颜色按亮度进行分类, 分成 256 个等级。不同比例的红、绿、蓝叠加, 能产生丰富的颜色。例如, 等比例的三原色进行相加可以产生白色, 红色与绿色相加产生黄色。可见, RGB 空间属于“叠加型”原色系统, 因此把 RGB 颜色空间作为最基础的颜色空间, 通过对 RGB 的非线性或线性变换可以获得其它的颜色空间。

##### 2) CIELAB 颜色空间

在许多文献中, CIELAB 颜色空间也称 CIE 1976  $L^*a^*b^*$ (简称为 CIE  $L^*a^*b^*$ ) 颜色空间。CIELAB 颜色系统是使用最广泛的物体颜色度量方法, 并作为度量颜色的国际标准。CIE 1976  $L^*a^*b^*$  颜色空间是 CIE 1931 XYZ 颜色空间的一种数学变换的结果。

CIE 1976  $L^*a^*b^*$  颜色空间和 CIE 1931 XYZ 颜色空间的相同之处是, 它们都使用相同的基本原理, 即颜色是光、物体和观察者组合的结果, 三种基色值是用 CIE 定义的光、物体和观察者的数据进行计算得到的。

CIELAB 系统使用的坐标叫做对色坐标 (opponent color coordinate), 使用对色坐标的想法来自这样的概念: 颜色不能同时是红和绿, 或者同时是黄和蓝, 但颜色可以被认为是红和黄、红和蓝、绿和黄以及绿和蓝的组合。CIELAB 使用  $L^*$ ,  $a^*$  和  $b^*$  坐标轴定义 CIE 颜色空间。其中,  $L^*$  值代表光亮度, 其值从 0(黑色) 100(白色)。 $a^*$  和  $b^*$  代表色度坐标, 其中  $a^*$  代表红-绿轴,  $b^*$  代表黄-蓝轴, 它们的值从 0 10。 $a^* = b^* = 0$  表示无色, 因此  $L^*$  就代表从黑到白的比例系数。

### 2.2 SIFT 特征

SIFT 特征 (Scale-invariant transform, 尺度不变特征变换) 由 David Lowe 在 1999 年 [7] 所发表, 2004 年 [8] 完善总结。它是一种计算机视觉的算法, 用来侦测与描述图像中的局部性特征, 它在空间尺度中寻找极值点, 并提取出其位置、尺度、旋转不变量。SIFT 算法提取的特征点具有尺度不变性,

也就是说，同一物体在图像上不论尺度大小，都能根据 SIFT 算法提取到相同的特征点。

简单来说，SIFT 算法就是用不同尺度 (标准差) 的高斯函数对图像进行平滑，然后比较平滑后图像的差别，差别大的像素就是特征明显的点。

尺度空间理论的目的是模拟图像数据的多尺度特征。高斯卷积核是实现尺度变换的唯一线性核，于是一幅二维图像的尺度空间定义为：

$$L(x, y, e) = G(x, y, e) * I(x, y) \quad (1)$$

其中  $G(x, y, e)$  是尺度可变高斯函数，

$$G(x, y, e) = \frac{1}{2 * \pi * e^2} * \exp\left[-\frac{x^2 + y^2}{2e^2}\right] \quad (2)$$

$(x, y)$  是空间坐标， $e$  是尺度坐标。

为了有效地在尺度空间检测到稳定的关键点，提出了高斯差分尺度空间 (DOG scale-space)。利用不同尺度的高斯差分核与图像卷积生成。

$$D(x, y, e) = ((G(x, y, ke) - G(x, y, e)) * I(x, y) = L(x, y, ke) - L(x, y, e) \quad (3)$$

SIFT 特征一般用于匹配，在 eye fixation prediction 中有用到，而在显著性区域检测中没有见用的。

### 3. 基于区域的 feature contrasts

这里的 feature contrasts 是指由某种 visual cue(比如颜色、纹理、位置等) 对图像中两个区域  $r_i$ 、 $r_j$  求出的区域间对比度。

#### 3.1 颜色

##### 3.1.1 Color Contrast

论文 [12] 中用到，公式如下：

$$C_i = \sum_{j=1}^n w(R_i) \Phi(i, j) ||c_i - c_j||_2 \quad (4)$$

其中,  $c_i$  和  $c_j$  分别是区域  $R_i$  和  $R_j$  内颜色的均值,  $w(R_j)$  表示区域  $R_j$  内的像素个数,  $\Phi(i, j) = \exp\{-D(R_i, R_j)/\sigma^2\}$ 。

论文 [14] 中用到, 图像区域中方差不同的颜色之间的分布是相互独立的。区域  $r_i$  与  $r_j$  之间的 color contrast 定义如下:

$$D_c(r_i, r_j) = \|\mu_{c,i} - \mu_{c,j}\|^2 \cdot \left( \frac{\sigma_{c,i}^2}{n_i} + \frac{\sigma_{c,j}^2}{n_j} \right)^{-\frac{1}{2}} \quad (5)$$

其中  $\mu_{c,i}$  和  $\mu_{c,j}$  分别表示区域  $r_i$  和  $r_j$  内的颜色均值,  $\sigma_{c,i}^2$  和  $\sigma_{c,j}^2$  表示方差,  $n_i$  和  $n_j$  表示相应区域内像素的个数。

### 3.1.2 颜色直方图

图像直方图是指统计图像中像素的灰度/颜色得到的图像灰度/颜色频数图。直方图由于其计算代价较小, 且具有图像平移、旋转、缩放不变性等优点, 广泛应用于图像处理的各个领域。Swain 和 Ballard 最先提出了使用颜色直方图作为图像颜色特征的代表方法。

传统颜色直方图描述方法存在以下问题:

- 1) 颜色特征维数高。以 8bit 的 RGB 颜色空间为例, 全颜色数为  $256 \times 256 \times 256$  种颜色, 如果以全颜色数统计直方图, 则存储空间和计算复杂度都较大。
- 2) 颜色特征受光照影响。即对于两幅颜色分布很类似却因光照不同导致亮度差异大的图像, 理论上, 其颜色直方图应相似, 但实际传统颜色直方图却不相似。
- 3) 不能表达相近颜色间相关性, 即传统颜色直方图的颜色间完全独立, 不能反映相近颜色间的关联。理论上, 对于发生较小颜色偏移的两幅图像间应相似。如, 一幅完全红色的图像与另一幅完全浅红色的图像间相似度较高。而实际传统颜色直方图却不相似。
- 4) 丢失空间位置信息, 因此该特征无法区分颜色相同而空间分布不同的两幅图像。

得到图像颜色特征后需要定义颜色特征的相似度量公式, 以表示两幅图像间颜色的相似性。不同的相似性度量公式对实际应用结果可能影响很大。因此需要研究如何选择或设计合适的相似性度量算法。

显著性 Models 中, CB [4]、DRFI [5]、HC/RC [1]、HDCT [6] 方法都用到了颜色直方图。

DRFI 方法中关于图像 RGB 空间的颜色直方图代码如下:

```
1 image = imread('3.jpg');
2 image_rgb = im2double( image );
3 RGB_bins = [16 16 16];
4 R = image_rgb(:,:,1);
```

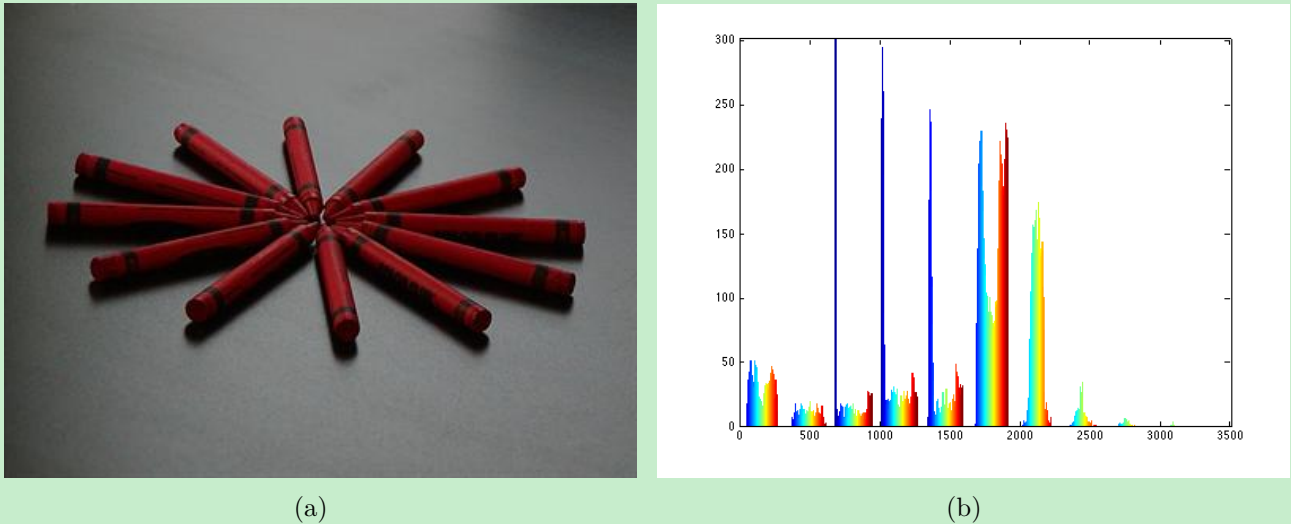


图 1: RGB 空间的颜色直方图

```

5 G = image_rgb(:,:,2);
6 B = image_rgb(:,:,3);
7 rr = min( floor(R*RGB_bins(1)) + 1, RGB_bins(1) );
8 gg = min( floor(G*RGB_bins(2)) + 1, RGB_bins(2) );
9 bb = min( floor(B*RGB_bins(3)) + 1, RGB_bins(3) );
10 Q_rgb = (rr-1) * RGB_bins(2) * RGB_bins(3) + ...
11         (gg-1) * RGB_bins(3) + ...
12         bb + 1;

```

首先对图像 ( $300 \times 400$ ) 的颜色空间进行量化, 将颜色空间划分为若干个小的颜色区间, 即直方图的 bin, 例如将每个颜色通道量化为只有 16 个不同值, 此时  $bin = 16 \times 16 \times 16$ , 然后计算矩阵  $Q(300 \times 400)$ , 用其中的值代表颜色, 而不是用  $(r, g, b)$  向量表示颜色,  $Q$  中有多少个不同值表示图像中有多少种颜色。结果如图 1。

对每个超像素区域, 可以知道其中像素点的坐标值, 能求出其对应到矩阵  $Q$  中的值, 有多少个不同值就代表该区域内有多少种不同的颜色, 然后算该区域在  $1 - 16^3$  之间的颜色占的像素个数, 没有的记为 0。对每个区域都能算出这样一个  $16^3$  维的向量, 然后求区域之间的直方图的对比度, 也就是求这样两个向量之间的距离。

### 3.2 纹理特征

图像纹理一直到现在都没有一个一致的、公认的定义, 它在图像中是一个重要但是又不太容易描述出来的特征。纹理是人们将人类的视觉与触觉联系起来, 进而形成一个视觉信息, 它起源于人类对事物的触感。



LBP(Local Binary Pattern, 局部二值模式) 首先是由 Ojala 等人 [10] 于 1994 年提出, DRFI 方法 [5] 中用到。

LBP 有很多变种, 或说改进。原始的 LBP 记录像素点与其周围像素点的对比信息, 或说差异。对于图像上 9 个方格中中间方格 (方格中的值是像素点灰度值大小), 做一个阈值化处理。大于等于中心点像素的, 标记为 1, 小于的则标记为 0。最后将中心像素点周围的 11110001 二进制数化为十进制数, 得到 LBP 值。如图 2 所示。

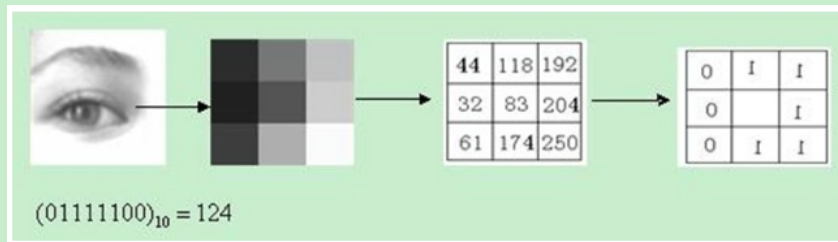


图 2: 原始 LBP

如图 3

### 3.3 位置特征

超像素区域中心与图像中心的距离

论文 [12] 中用到, 公式如下:

$$H_i = \frac{1}{w(R_i)} \sum_{x_i \in R_i} \exp\{-\lambda \|x_i - x_c\|^2\} \quad (6)$$

其中  $(x_0, x_1 \dots)$  是区域  $R_i$  中的像素坐标集,  $x_c$  是图像中心的坐标,  $w(R_i)$  计算了区域  $R_i$  内的像素个数。由  $H_i$  的公式可看到, 距离图像中心越近的区域拥有越大的权值。

HDCT [6] 中用到,

### 3.4 形状特征

#### 3.4.1 区域面积

该区域包含的超像素的个数

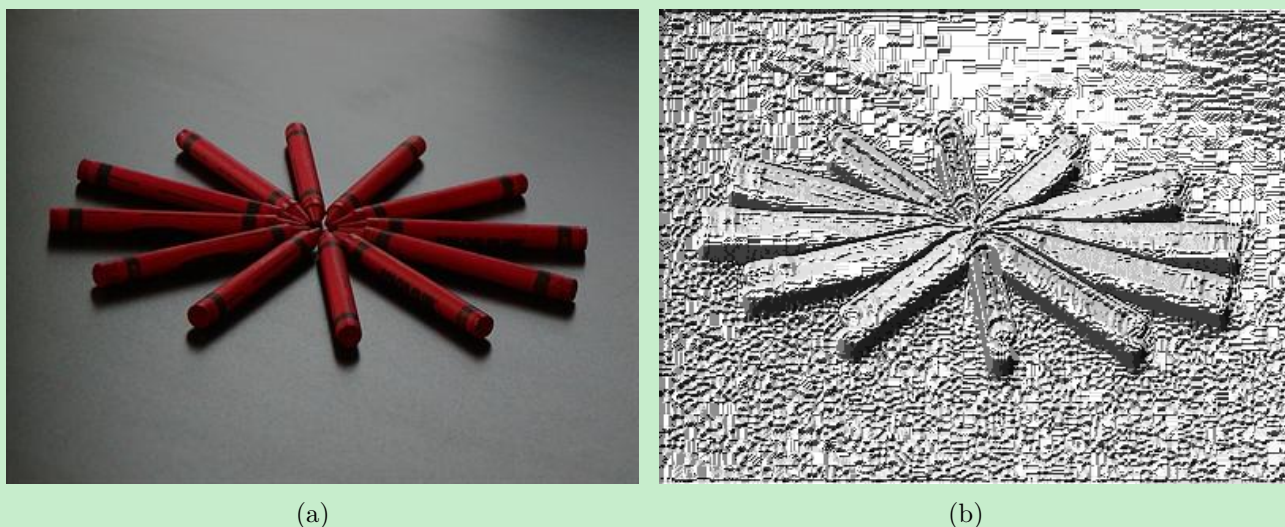


图 3: LBP

### 3.4.2 奇异值特征

奇异值特征 (Singular Value Feature, SVF) [11] 被用来从测试图像中检测模糊区域, 通常一幅图像中的模糊区域是背景的可能性较大。

### 3.4.3 HOG 特征

HOG(Histogram of Oriented Gradients, 方向梯度直方图) 特征最早是由法国国家计算机技术和控制研究所 (INRIA) 的 Navneet Dalal 和 Bill Triggs 在 2005 年发表在 CVPR 上的论文 [2] 中提出的。

Dalal 提出的 HOG 特征提取的过程如图 4, 进一步表述如下:

1. 灰度化

2. 采用 Gamma 校正法对输入图像进行颜色空间的标准化 (归一化); 目的是调节图像的对比度, 降低图像局部的阴影和光照变化所造成的影响, 同时可以抑制噪音的干扰。Gamma 压缩公式:

$$I(x, y) = I(x, y)^{\gamma} \quad (7)$$

3. 计算图像每个像素的梯度 (包括大小和方向), 主要是为了捕获轮廓信息, 同时进一步弱化光照的干扰。

图像中像素点  $(x, y)$  的梯度为

$$G_x(x, y) = H(x + 1, y) - H(x - 1, y) \quad (8)$$

$$G_y(x, y) = H(x, y + 1) - H(x, y - 1) \quad (9)$$



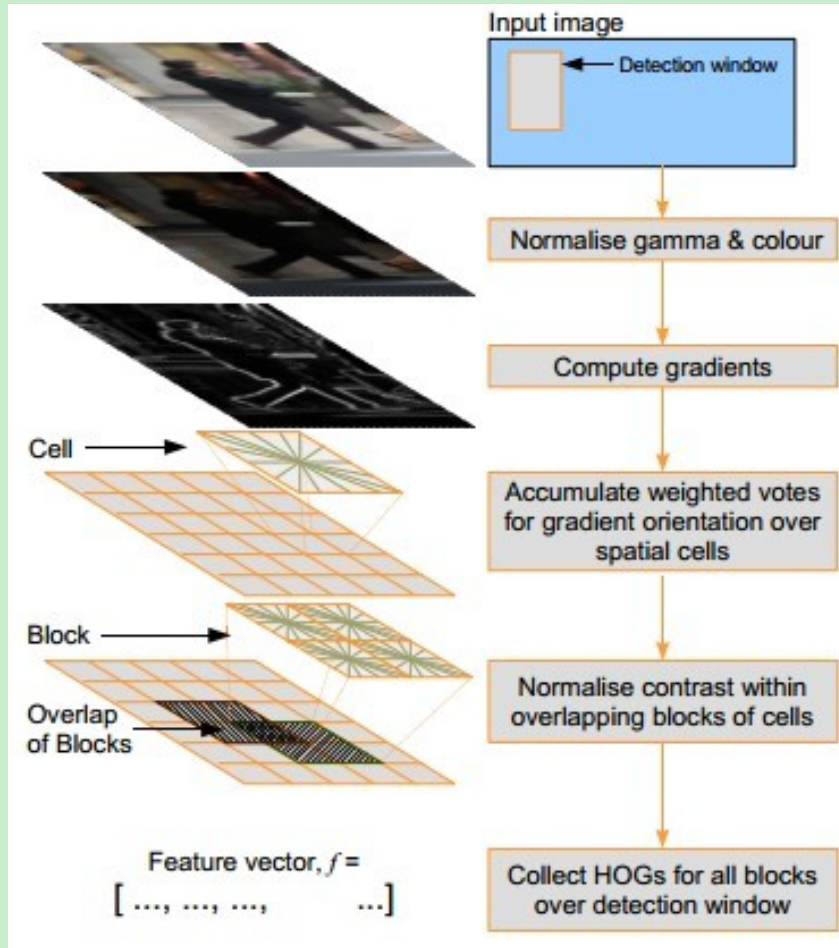


图 4: 算法流程图

式中  $G_x(x, y)$ ,  $G_y(x, y)$ ,  $H(x, y)$  分别表示输入图像中像素点  $(x, y)$  处的水平方向梯度、垂直方向梯度和灰度值。像素点  $(x, y)$  处的梯度幅值和梯度方向分别为：

$$G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \quad (10)$$

$$\alpha(x, y) = \tan^{-1}\left(\frac{G_y(x, y)}{G_x(x, y)}\right) \quad (11)$$

最常用的方法是：首先用  $[-1, 0, 1]$  梯度算子对原图像做卷积运算，得到  $x$  方向（水平方向，以向右为正方向）的梯度分量  $\text{gradscal}_x$ ，然后用  $[1, 0, -1]^T$  梯度算子对原图像做卷积运算，得到  $y$  方向（竖直方向，以向上为正方向）的梯度分量  $\text{gradscal}_y$ 。然后再用以上公式计算该像素点的梯度大小和方向。

4. 将图像每  $16 \times 16$  (取其它也可以) 个像素分到一个 cell 中，对于  $256 \times 256$  的图像来说，就分成了  $16 \times 16$  个 cell 了。

5. 对于每个 cell 求其梯度方向直方图，通常取  $\text{bin} = 9$  (取其它也可以) 个方向 (特征)，也就是每  $360/9 = 40$  度分到一个方向，形成每个 cell 的 descriptor。如图 5，当某像素的梯度方向是 20-40 度，

然后它的梯度大小是  $|grad|$ ，那么直方图第 2 个 bin 的计数就要加上  $|grad|$ 。

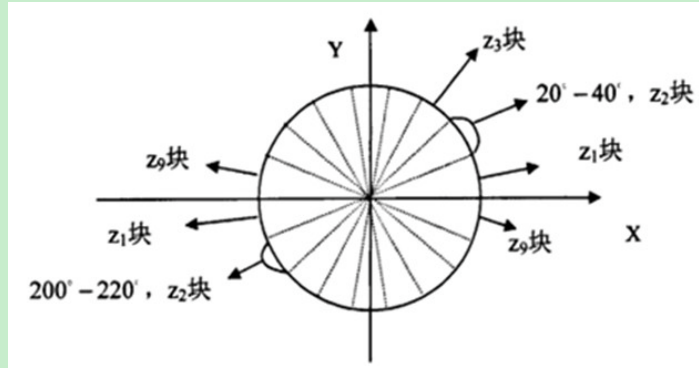


图 5: 梯度方向 bin 的划分

6. 由于局部光照的变化以及前景 - 背景对比度的变化，使得梯度强度的变化范围非常大。这就需要对梯度强度做归一化。归一化能够进一步地对光照、阴影和边缘进行压缩。为此可以将每  $2 \times 2$  (取其它也可以) 个 cell 合成一个大的、空间上连通的 block，所以这里就有  $(16 - 1) \times (16 - 1) = 225$  个 block。一个 block 内所有 cell 的特征向量串联起来便得到该 block 的 HOG 特征。这些 block 是互有重叠的，这就意味着：每一个 cell 的特征会以不同的结果多次出现在最后的特征向量中。

7. 所以每个 block 中都有  $2 \times 2 \times 9$  个特征，一共有 225 个 block，所以总的特征有  $225 \times 36$  个。

遇到的疑问：

- 1) 在显著性检测中，先将图像分割成区域，然后怎么提取该区域的 HOG 特征？
- 2) 一定要将彩色图像先转化为灰度图像才能提取 HOG 特征吗？

HDCT [6] 方法中用到 HOG 特征 [3]，采用的是 VLFeat 官网上的代码。论文中首先也是对图像进行超像素分割，然后求每个超像素区域内像素点坐标的平均值  $(x_i, y_i)$ ， $i$  代表第  $i$  个超像素区域，求该区域的 HOG 特征就是以该坐标点为中心的  $17 \times 17$  的网格作为输入图像 (只取该网格内图像  $r$  通道的值)，17 作为 cellSize，然后对每个超像素区域得到一个 31 维的 HOG 特征向量。

### 3.5 Visual Complexity Contrast

论文 [14] 中用到，信息论中可以用熵来计算 visual complexity，区域  $r_i$  和  $r_j$  之间的 visual complexity contrast  $D_e(r_i, r_j)$  就被定义为

$$D_e(r_i, r_j) = [H(r_i) - H(r_j)]^2 \quad (12)$$

其中,  $H(r_i)$  表示区域  $r_i$  内的熵

$$H(r_i) = \sum_{p=1}^{n_{c,i}} f(c_{p,i}) \cdot \log_2 f(c_{p,i}) \quad (13)$$

其中,  $c_{p,i}$  是区域  $r_i$  中第  $p$  种颜色,  $n_{c,i}$  区域  $r_i$  中包含的颜色个数,  $f(c_{p,i})$  表示区域  $r_i$  中颜色  $c_{p,i}$  出现的概率。

### 3.6 Background Weighted Contrast

论文 [15] 中用到

## 4. Feature Contrasts 的融合

求出两个区域间关于多种关于 visual cues 的对比度之后, 要先将这些对比度融合, 例如可以相乘, 求出最终的对比度公式。然后再对每个区域计算 local 或 global contrast。

论文 [14] 中, 两个区域  $r_i$ 、 $r_j$  间的最终的对比度定义如下:

$$D_r(r_i, r_j) = D_c(r_i, r_j) \cdot \exp[\sigma_e^2 \cdot D_e(r_i, r_j)] \quad (14)$$

其中  $D_c(r_i, r_j)$  是指 color contrast,  $D_e(r_i, r_j)$  是指 visual complexity contrast。

对每个区域  $r_i$ , 其空间加权 global contrast 定义为:

$$U(r_i) = \sum_{j \neq i} w_{ij} \cdot D_r(r_i, r_j) \cdot \phi_j \quad (15)$$

$$w_{ij} = \frac{1}{Z_i} \cdot \exp[-\sigma_s^2 \cdot D_s(r_i, r_j)] \quad (16)$$

其中,  $\phi_j$  是指区域  $r_j$  内的像素个数, 即区域  $r_j$  的大小。  $D_s(r_i, r_j)$  表示区域  $r_i$  和  $r_j$  之间的空间距离。  $\phi_s$  则用来控制空间加权  $w_{ij}$  的影响程度,  $\phi_s$  越大, 对  $U(r_i)$  的影响越小。  $\frac{1}{Z_i}$  是归一化因子, 保证  $\sum_{j \neq i} w_{ij} = 1$ 。

## 5. Priors

### 5.1 Backgroundness

计算超像素区域与图像四个边界上的超像素的相似度。

论文 [9] 中用到

### 5.2 Center Prior 或 Location Prior

SDSP [13] 中将 Location Prior 描述为：处于图像偏中央位置的物体更能吸引人的注意。这里 Location Prior 是按像素级计算的，定义如下：

$$S_D(x) = \exp\left(-\frac{\|x - c\|_2^2}{\sigma_D^2}\right) \quad (17)$$

效果如图 6。

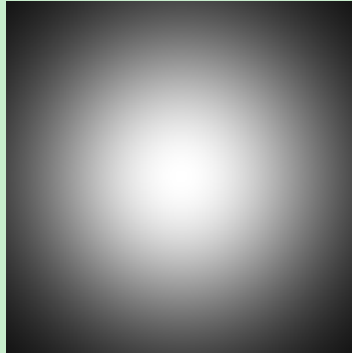


图 6: SDSP: Center Prior

### 5.3 Color Prior

SDSP [13] 中将 Color Prior 描述为：暖色（例如红色和黄色）比冷色（例如绿色和蓝色）更能吸引人类视觉系统的注意。先将图像由 RGB 颜色空间转换到  $CIE L^*a^*b^*$  空间， $\{f_L(x)\}$ ,  $\{f_a(x)\}$ ,  $\{f_b(x)\}$  分别代表  $L^*$  通道， $a^*$  通道和  $b^*$  通道。这里 Color Prior 是按像素级计算的，定义如下：

$$S_c(x) = 1 - \exp\left(-\frac{f_{an}^2(x) + f_{bn}^2(x)}{\sigma_c^2}\right) \quad (18)$$

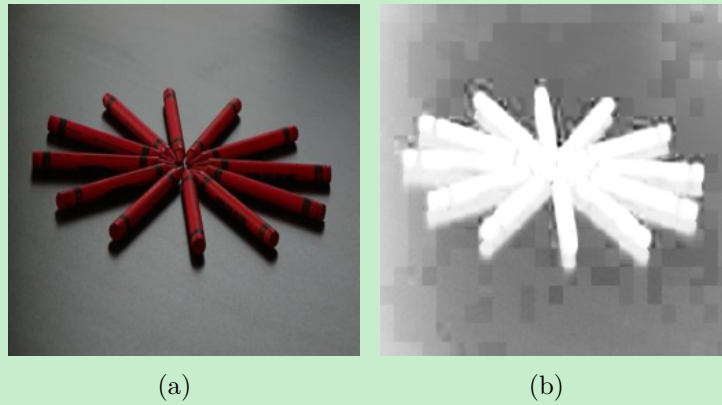


图 7: SDSP: Color Prior

其中

$$f_{an}(x) = \frac{f_a(x) - \min a}{\max a - \min a}, f_{bn}(x) = \frac{f_b(x) - \min b}{\max b - \min b} \quad (19)$$

效果如图 7。

## 参考文献

- [1] Ming-Ming Cheng, Guo-Xin Zhang, Niloy J Mitra, Xiaolei Huang, and Shi-Min Hu. Global contrast based salient region detection. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 409–416. IEEE, 2011.
- [2] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.
- [3] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.
- [4] Huaizu Jiang, Jingdong Wang, Zejian Yuan, Tie Liu, Nanning Zheng, and Shipeng Li. Automatic salient object segmentation based on context and shape prior. In *BMVC*, volume 6, page 7, 2011.



- [5] Huaizu Jiang, Jingdong Wang, Zejian Yuan, Yang Wu, Nanning Zheng, and Shipeng Li. Salient object detection: A discriminative regional feature integration approach. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2083–2090. IEEE, 2013.
- [6] Jiwhan Kim, Dongyoon Han, Yu-Wing Tai, and Junmo Kim. Salient region detection via high-dimensional color transform. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 883–890. IEEE, 2014.
- [7] David G Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.
- [8] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
- [9] Song Lu, Vijay Mahadevan, and Nuno Vasconcelos. Learning optimal seeds for diffusion-based salient object detection. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 2790–2797. IEEE, 2014.
- [10] Timo Ojala, Matti Pietikainen, and David Harwood. Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In *Pattern Recognition, 1994. Vol. 1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on*, number 1, pages 582–585, 1994.
- [11] Bolan Su, Shijian Lu, and Chew Lim Tan. Blurred image region detection and classification. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1397–1400. ACM, 2011.
- [12] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 1155–1162. IEEE, 2013.
- [13] Lin Zhang, Zhongyi Gu, and Hongyu Li. Sdsp: A novel saliency detection method by combining simple priors. In *ICIP*, pages 171–175. Citeseer, 2013.
- [14] Guokang Zhu, Qi Wang, and Yuan Yuan. Tag-saliency: Combining bottom-up and top-down information for saliency detection. *Computer Vision and Image Understanding*, 118:40–49, 2014.

- [15] Wangjiang Zhu, Shuang Liang, Yichen Wei, and Jian Sun. Saliency optimization from robust background detection. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 2814–2821. IEEE, 2014.