

一种新的基于图论的图像匹配算法

伏雪， 马燕， 林涛

(上海师范大学 计算机科学与工程系, 上海, 中国, 200234)

摘要: SIFT 算法在图像匹配领域中占有重要地位, 但是, 利用 SIFT 算法提取的图像特征点, 是分布在整幅图像中的, 这就造成提取的特征点不集中。本文结合图论的方法, 对 SIFT 算法提取的特征点进行处理, 去除部分不集中的点, 从而达到提高匹配效率的目的, 我们将该方法命名为 G-SIFT 算法。G-SIFT 算法提取的每一个特征点视为图的顶点, 将这些顶点的一元关系视为图的边, 并利用这些边的大小特点对特征点进行处理, 使得处理后的特征点主要集中在物体上。实验证明, 利用该方法对图像进行处理后, 特征点匹配结果更加集中, 匹配率最高提高了 1.4%, 匹配点集中在物体上的正确率最高提高了 9.1%。

关键字: SIFT; G-SIFT; 图论方法; 特征点匹配

中图分类号: TP391. 41

文献标志码: A

A Novel Image Matching Algorithm Based on Graph Theory

FU Xue, MA Yan, LIN Tao

(Department of Computer Science and Technology, Shanghai Normal University, Shanghai, 200234, China)

Abstract: SIFT algorithm plays an important role in the field of image matching. However, the feature points extracted by SIFT algorithm distributing among the whole image. This situation result in the feature points are not concentrated. This paper developed a new G-SIFT algorithm based on the SIFT algorithm. The G-SIFT algorithm combines the graph theory with the SIFT algorithm, removing the SIFT feature points, which are not concentrated. In the graph theory, every feature point is treated as a vertex. The unary terms of these vertices is taken as edges of graph. The feature points are processed according to those edges. The experiments show that the feature points matching results are more concentrated and concentration rate is improved by 12%.

Keywords: SIFT; G-SIFT; Graph theory; Feature point matching

0 引言

基于尺度空间的特征匹配算法 (SIFT) 是 David G. Lowe 提出来的, 并在图像处理中广泛应用^[1, 2]。SIFT 算法提取的特征点对尺度、旋转、平移的不变性, 而且对光照变化、仿射变换和 3 维投影变换具有一定的鲁棒性, 因此广泛应用于图像匹配^[1, 3]和图像检索^[4, 5]等领域。传统的 SIFT 算法主要是针对整幅图像进行处理, 在利用 SIFT 算法对图像进行匹配时, 匹配成功的点中包含大量的背景信息点。同时, 由于每个 SIFT 特征点都有 128 维特征向量, 这无疑也加大了匹配过程中的工作量。基于此, 许多研究者在 SIFT 算法的基础上提出了 SIFT 改进算法。相关研究者提出了兴趣区

域, 并将其应用到图像处理中, 使在对图像进行处理过程中目标更加明确, 从而实现图像处理的效率^[6]。

Feng-Xiang GE^[7]等人结合 PCA-SIFT 算法降低 SIFT 特征描述子的向量维数, 从而提高图像处理的速度。本文结合图论方法, 对 SIFT 算法进行改进, 使得特征点匹配结果更加集中。

本研究主要贡献为: 在保留 SIFT 特征点原有的优点之后, 结合图论方法, 产生的新的描述子在特征点匹配的过程中更加集中。同时该方法有效的去除了图像中的背景点及不集中的点, 能够有效地提高

基金项目: 国家自然科学基金资助项目 (编号: 61373004)

作者简介: 伏雪 (1990-), 女, 硕士研究生, 研究方向: 图像处理与模式识别; 马燕 (1970-), 女, 教授, 博士, 研究方向: 图像处理与模式识别; 林涛 (1988-), 男, 硕士研究生, 研究方向: 图像处理。

Email: fuxuee@hotmail.com

SIFT 算法在图像检索中的效率。文章中试验采用的图像主要是针对办公室场景下的图片。

1 SIFT 算法

Lowe 提出了基于模板图像目标检测的 SIFT 算法。SIFT 算法产生局部描述子，对平移、尺度变换和旋转都保持不变性。这些描述子对光照变化和三维投影也保持局部不变性。Cui, Chunhui^[8]等人 将 SIFT 描述只和其他特征不变描述子进行了对比。并总结除了 SIFT 描述子在尺度、旋转和光照条件改变后能够更好的保持不变性。

关键点检测的第一个阶段就是确定关键的位置和尺度。对于图像尺度变换保持不变的位置检测，可以使用一个成为尺度空间的连续的尺度函数，通过搜索所有尺度中的稳定特征来实现。尺度空间定义为 $L(x, y, \sigma)$ ，由高斯函数与输入图像 $I_b(x, y)$ 卷积产生的，如公式(1)所示。

$$L(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) * I_b(x, y) \quad (1)$$

公式中*是高斯运算符，(x 、 y)代表图像上的点， σ 是尺度因子。

为了有效的检测尺度空间中稳定的关键点位置，Lowe 提出使用空间极值点检测的方法。空间极值点是由相邻的两个空间产生的高斯差分图像计算得来的。相邻的两个空间是由一个常量因子 k 决定的，如公式(2)所示。

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (2)$$

为了检测极值 $D(x, y, \sigma)$ ，每一个采样点都需要跟同

一平面中的其他 8 个像素以及相邻尺度空间中该像素所对应位置的 9×2 个像素进行计算比较，只有被检测的采样点的高斯差分数值都大于或都小于与它进行比较的 26 个像素的高斯差分数值时，才能把此点作为一个局部极值点。上述方法得到的极值点中存在低对比度的点或者不稳定的边缘上的一些点，这些点将会被去除。

利用 SIFT 算法产生的关键点描述子是由采样的大小和关键点周围图像块的梯度方向决定的。最终形成 128 维的 SIFT 特征向量就是特征描述符。SIFT 算法提取的特征点描述子定义为 $\text{des}(i, j)$ ，特征点信息为 $\text{locs}(i, j)$ 。这些描述子被归一化到单位长度，因此对光照保持不变性。

利用上述方式生成 SIFT 特征响亮后，我们选取关键点特征向量的欧氏距离作为判定相似性的度量。第一，以一幅图像为标准，取出它的一个特征点，进而对另一张图中的特征点进行搜索，记录前两个欧氏距离最近的点。第二，对上一步产生的欧氏距离进行比较，先确定一个阙值，若最小的距离除次小的距离的结果小于这一值，则认为这两个特征点匹配成功，否则就不匹配。

2 图论方法

2.1 图表示

为了 SIFT 算法提取的特征点更集中，本文中结合图论方法^[9, 10]，对 SIFT 算法进行改进，并将其命名为 G-SIFT 算法。

具体流程如图 1 所示：

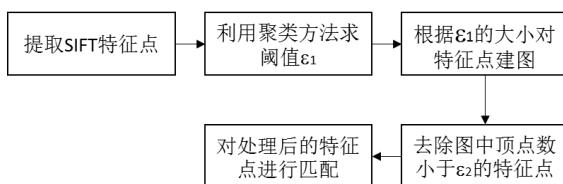


图 1 流程图

具体算法步骤如下：

- 1) 获取 SIFT 特征点 $\{n_1, n_2, \dots, n_N\}$;
- 2) 利用 K-means 求得特征点之间的平均距离 ϵ_1 ;
- 3) 对特征点建图，以特征点 n_i 为例

```

for j = 1:N
    if 特征点 n_i 和 n_j 的距离小于 ε₁
        line (n_i, n_j), n_i 的度 D_i 加 1
    else
    end
  
```

从而得到点 n_i 的图 $n_i = \sum_{j=1}^N E(i, j)$ 。

- 4) 去除特征点，以特征点 n_i 为例

```

if D_i 大于 ε₂
    保留点 n_i
else
    去除点 n_i
end
  
```

- 5) 对处理后的特征点进行匹配

在图论方法中，图 G 表示为 $G = (V, E)$ ，在这里 V 表示定点， E 表示边矩阵代表着图中各定点之间的关系。我们将图像提取的关键点定义为 (v) ，图像中图表形式分别是 $G(V, E)$ 。其中， $V = \{v_i\}, i = 1, \dots, I$ ，

E 是一个 $I \times I$ 的矩阵，定义如公式(3)所示：

$$E(i, j) = \begin{cases} d_{ij}, & \text{if } d_{ij} < \epsilon_1 \\ 0, & \text{otherwise} \end{cases}$$

$$(3) \quad \text{在这里 } d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}。 \epsilon_1 \text{ 表示两个特征点之间的距离阈值。} \epsilon_1 \text{ 的取值根据如下阈值计算得出：1、利用 k 均值算法对特征点进行聚类；2、求出每一类中所有特征点到聚类中心的平均距离；3、求所有类的平均距离。并将其值赋予 } \epsilon_1。 E(i, j) = 0 \text{ 表示点 } v_i \text{ 与 } v_j \text{ 之间没有边。}$$

2.2 K 均值概述

K-均值算法的基本思想：把 n 个具有 w 个维数的数据样本分成 K 类。聚类的数目 K 需要事前设定，并从数据集中随机选择 K 个数据作为聚类的初始中心。然后计算每个数据对象与中心的距离，并根据最小的距离值将数据放到离其最近的簇，当每个数据点都分配到自己所属的簇后重新计算每个簇的新的聚类中心。反复迭代以上两步，直到满足收敛条件为止^[11]。

K 均值算法需要用户给定簇的个数和数据样本个数、维数等参数，然后对数据集进行聚类。K 均值的算法流程为：

- (1) 从含有 n 个数据对象的数据集中选择 K 个数据对象作为初始的聚类中心；
- (2) 计算每个数据对象与中心的距离，并根据最小的距离值将数据放到离其最近的簇，并计算误差平方和准则函数的值；
- (3) 当每个数据点都分配到自己所属的簇后重新计算每个簇的新的聚类中心；
- (4) 将步骤 (3) 计算得到的误差平方和准则函数的值和上一次迭代计算得到的值进行比较，若两者差值小于预先设定的收敛阈值，即满足收敛条件，转到

步骤(5)，否则转到步骤(2)；

(5) 输出 K 均值聚类的结果，到此该算法结束。

本文主要利用 k 均值算法对特征点进行聚类；然后，求出每一类特征点到聚类中心的平均距离；最后，得出特征点之间的平均距离，作为建立图的阈值。

2.3 去除特征点

接下来，文章将论述如何去除背景上及其他物体上的无用的匹配点。利用图论方法获得的矩阵后，接下来我们要根据矩阵代表的信息对 SIFT 算法提取的特征点描述子及特征点信息进行处理，文章中将图论方法改进后的 G-SIFT 算法提取的特征点描述子命名为 gdes(i, j) 将经特征点的信息表示为 glocs(i, j)，在对 gdes(i, j) 和 glocs(i, j) 进行定义之前，要先对原 SIFT 描述子及其信息进行如下处理。如公式(4)和公式(5)所示

$$\text{des}(i,:) = \begin{cases} 0, & \text{if } \text{count}(E(i,:)) < \varepsilon_2 \\ \text{des}(i,:), & \text{otherwise} \end{cases} \quad (4)$$

$$\text{locs}(i,:) = \begin{cases} 0, & \text{if } \text{count}(E(i,:)) < \varepsilon_2 \\ \text{locs}(i,:), & \text{otherwise} \end{cases} \quad (5)$$

在这里， $\text{count}(E(i,:))$ 表示矩阵 E 中的 i 行 j 列中存在边的个数，的大小根据图像中特征点的密集度来决定， ε_2 的取值不同会对去点的效果产生影响。本文中 ε_2 分别去 2、3、4，在试验结果中我们会对比三个不同取值对实验结果的影响。

根据处理后的 des(i, j) 和 locs(i, j)，我们利用如下公式得到新的描述子及信息 gdes(i, j) 和 glocs(i, j)，如公式(6)和(7)所示：

$$\text{gdes}(k,:) = \begin{cases} \text{des}(i,:), & \text{if } \text{sum}(\text{des}(i,:)) = 0 \\ \text{gdes}(k,:), & \text{otherwise} \end{cases} \quad (6)$$

$$\text{glocs}(k,:) = \begin{cases} \text{locs}(i,:), & \text{if } \text{sum}(\text{locs}(i,:)) = 0 \\ \text{glocs}(k,:), & \text{otherwise} \end{cases} \quad (7)$$

接下来，要对根据图论方法处理后得到的新的描述子进行匹配，匹配方法与原 SIFT 算法一样。用两个特征点之间的距离及阈值，来判断两特征点是否匹配。

3 实验结果及分析

实验环境为 CPU3.20GHz，内存为 4.0Gbyte，操作系统为 Windows8.1，仿真平台为 MATLAB2014a。本文采用了 Caltech-256 图像库中的背景图像库集^[12]，选取了其中的六图像，每幅图像大小均为 896×592，实验利用两幅图片进行对比，分别用传统的 SIFT 算法和基于图论的改进后的 G-SIFT 算法对图像进行处理，然后从匹配的点数和匹配点集中区域进行比较。我们实验总共分为两部分，第一部分，结合两种算法对多幅办公室场景图片的匹配结果进行对比分析；第二部分，取其中两幅图片，分别设定不同的 ε_2 的值，对 G-SIFT 算法的结果进行对比。

3.1 第一部分

实验中，我们选取了办公室场景中的六张图片，如图 2 所示，图 2 总共有 5 列，第一列为原始图像，第二列为 SIFT 算法提取的特征点，第三列为 SIFT 特征点匹配结果，第四列为 G-SIFT 特征点，第五列为 G-SIFT 特征点匹配结果。从实验结果的图中我们可以看到 G-SIFT 算法很好的实现了去除 SIFT 特征点，达到了提高匹配精确度的目的。表 1 对应图 2 的图片，将两种算法分别从去除的点数、匹配率和正确匹配率三个方面进行了对比分析。根据表 1 可以总结出，G-SIFT 算法很好的实现了提高匹配率及使匹配点能

够很好的集中在物体上，在减少匹配点的情况下将匹配率最高提高了 1.4%，匹配点集中在物体上的正确率最高提高了 9.1%。在本文中，计算匹配率和匹配正确率方式分别如公式(8)和公式(9)所示：

$$\text{匹配率} = \frac{\text{匹配的点数}}{\text{两幅图像中提取的特征点总和}} \quad (8)$$

$$\text{匹配正确率} = \frac{\text{正确匹配的点数}}{\text{匹配的总点数}} \quad (9)$$

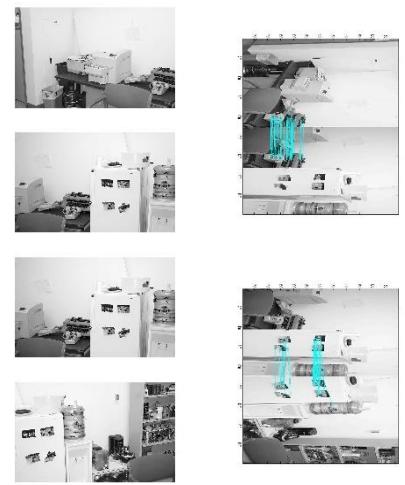
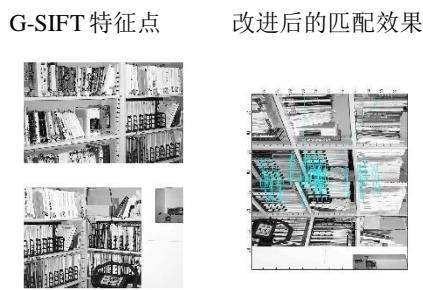
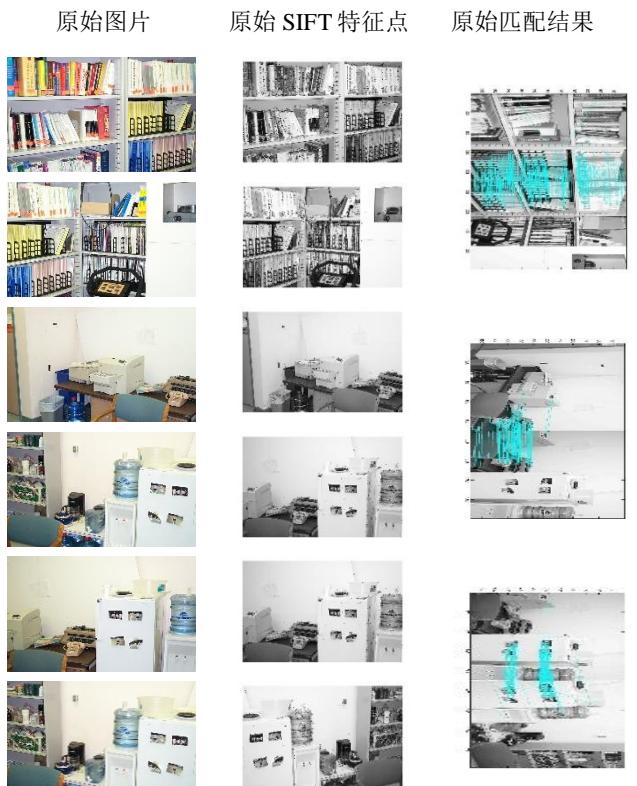


图 2 SIFT 算法及 G-SIFT 算法对图像处理结果比较

表 1 SIFT 算法及 G-SIFT 算法匹配结果分析



特征点数	匹配的点数	匹配率	匹配正确率
4895	358	4.45%	97.2%
3144			
830	134	6.62%	98.5%
1195			
1192	99	3.31%	90.9%
1803			

特征点数	匹配点数	匹配率	匹配正确率
2137	145	4.63%	98.3%
992			
357	79	7.49%	100%
712			
712	72	4.10%	100%
1044			

3.2 第二部分

在第二部分的试验中，选取了其中两幅图片， ε_2 分别取值为 2、3、4，对实验的结果进行对比。实验结

果分别如图 3 和表 2 所示。图 3 前两行第二列为原
始图像，第三列和第四列分别为 SIFT 算法的特征点及
匹配结果；第三、四行的第四列和第五列分别为 ϵ_2 取
2 的情况下 G-SIFT 算法提取的特征点及匹配结果；第
五、六行的第四列和第五列分别为 ϵ_2 取 3 的情况下
G-SIFT 算法提取的特征点及匹配结果；第七、八行的
第四列和第五列分别为 ϵ_2 取 4 的情况下 G-SIFT 算法
提取的特征点及匹配结果。表 2 对应于图 3 的图片，
对不同阈值下两种算法处理图片的结果进行了分析。
从结果中可以看出，不同阈值下 G-SIFT 算法对图像的
匹配率和正确匹配率有不同的影响。在本文选取的办
公室场景图片中，当 ϵ_2 取 3 时，G-SIFT 算法对匹配率
和正确匹配率的效果最好。

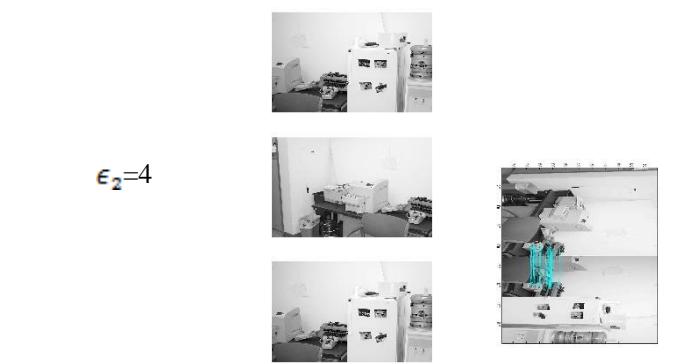
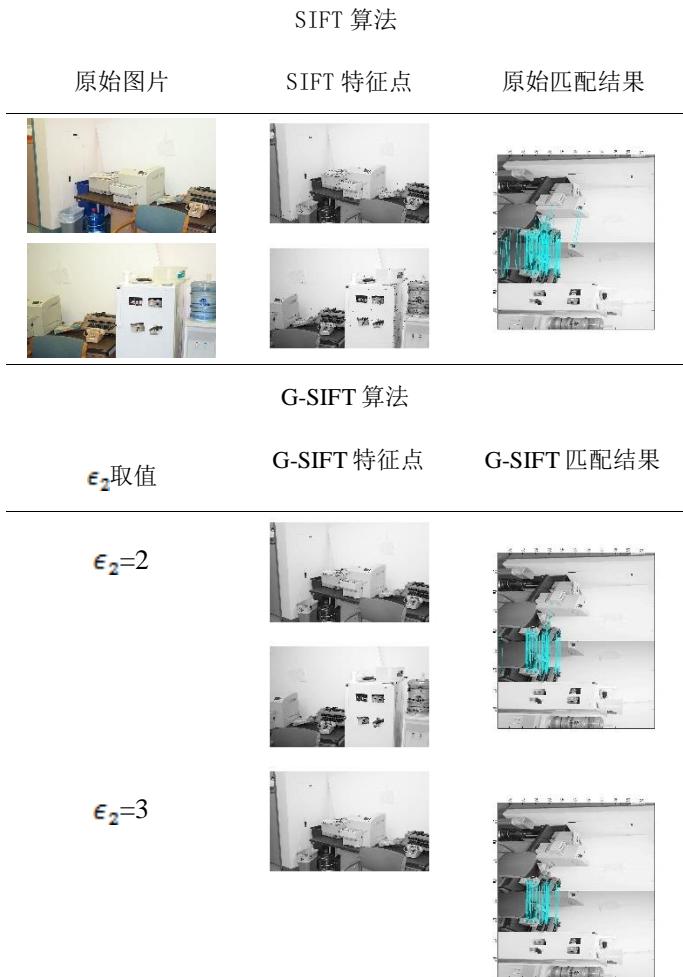


图 3 不同阈值下两种算法匹配效果对比图

表 2 不同阈值下两种算法统计结果对比

SIFT				
特征点数	匹配点数	匹配率	正确匹配率	
830	134	6.62%	98.5%	
1195				
G-SIFT				
ϵ_2 取值	特征点数	匹配点数	匹配率	正确匹配率
$\epsilon_2=2$	401	87	7.53%	99%
	753			
$\epsilon_2=3$	357	79	7.49%	100%
	712			
$\epsilon_2=4$	219	50	6.89%	100%
	507			

4 总结

本位提出的 G-SIFT 算法，保留了 SIFT 描述子对
尺度、旋转和光照条件改变的鲁棒性。同时结合图论
方法，有效的去除了特征点中无用的背景点及非目标
物体上的点，提高了匹配率及匹配点在物体上的集中

率。实验结果表明，此算法在办公室场景的图片取得了很好的实验效果。

参考文献

- [1] Joglekar J, Gedam S S, Mohan B K. Image Matching Using SIFT Features and Relaxation Labeling Technique—A Constraint Initializing Method for Dense Stereo Matching[J]. IEEE Transactions on Geoscience and Remote Sensing, 2014,52(9):5643-5652.
- [2] Tianze C, Limin C. A Union Matching Method for SAR Images Based on SIFT and Edge Strength[J]. Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of, 2014,7(12):4897-4906.
- [3] Dellinger F, Delon J, Gousseau Y, et al. SAR-SIFT: A SIFT-Like Algorithm for SAR Images[J]. Geoscience and Remote Sensing, IEEE Transactions on, 2015,53(1):453-466.
- [4] Wengang Z, Houqiang L, Richang H, et al. BSIFT: Toward Data-Independent Codebook for Large Scale Image Search[J]. Image Processing, IEEE Transactions on, 2015,24(3):967-979.
- [5] Feng W, Hongjian Y, Xingyu F. Adapted Anisotropic Gaussian SIFT Matching Strategy for SAR Registration[J]. Geoscience and Remote Sensing Letters, IEEE, 2015,12(1):160-164.
- [6] Zhang L, Qiu B, Yu X, et al. Multi-scale hybrid saliency analysis for region of interest detection in very high resolution remote sensing images[J]. Image and Vision Computing, 2015,35(0):1-13.
- [7] Feng-Xiang G, Yishu S, Bo S, et al. Sparse representation based classification by using PCA-SIFT descriptors: Information Science and Technology (ICIST), 2014 4th IEEE International Conference on, Shenzhen, 2014[C].2014.
- [8] Chunhui C, King N N. Scale- and Affine-Invariant Fan Feature[J]. Image Processing, IEEE Transactions on, 2011,20(6):1627-1640.
- [9] Pawar V, Zaveri M. Graph based K-nearest neighbor minutiae clustering for fingerprint recognition: 10th International Conference on Natural Computation (ICNC), Xiamen, 2014[C].2014.
- [10] Abusham E E A, Bashier H K. Face recognition using local graph theory (LGT): Computing, Electrical and Electronics Engineering (ICCEEE), 2013 International Conference on, Khartoum, 2013[C].2013.
- [11] Kaibing Z, Dacheng T, Xinbo G, et al. Learning Multiple Linear Mappings for Efficient Single Image Super-Resolution[J]. Image Processing, IEEE Transactions on, 2015,24(3):846-861.
- [12] <http://www.vision.caltech.edu/html-files/archive.html>