

一种带有“遮罩”的 ASIFT 特征提取算法

朱 博 戴先中 李新德 杨 伟

(东南大学自动化学院复杂工程系统测量与控制教育部重点实验室 南京 210096)

摘 要 指出提高 ASIFT 算法效率的一种新途径:利用“遮罩”屏蔽源自视角模拟的无效像素区域,从而使较为耗时的区域运算仅作用于有效像素区域,以达到提高运算效率的目的.首先,分析无效像素区域产生根源,并定量分析此类区域在各区域运算环节占有的比例,指出增加“遮罩”的必要性,特别指出被处理图像的宽高比越大,则经“遮罩”处理后所获得的加速比越高;随后,给出带有“遮罩”的 ASIFT 算法结构;然后,给出“遮罩”的一种低存储空间、高访问效率的数据结构,并给出离线获取“遮罩”的一种解析方法;最后,通过实验证实使用“遮罩”能有效提高 ASIFT 算法中区域运算部分的效率,从而在一定程度上也提高了算法整体的运算效率,但效率提升幅度受到理想区域加速比限制.

关键词 ASIFT 算法;完全仿射不变;特征提取;加速算法;计算效率

中图法分类号 TP391 **DOI 号** 10.11897/SP.J.1016.2015.01202

An ASIFT Algorithm with Masks for Feature Extraction

ZHU Bo DAI Xian-Zhong LI Xin-De YANG Wei

(Key Laboratory of Measurement and Control of CSE, School of Automation, Southeast University, Nanjing 210096)

Abstract A novel way is proposed to improve the efficiency of ASIFT algorithm in this paper; masks are used to shield invalid pixel areas which are caused by view simulation; thus the area computing part of ASIFT, which is more time-consuming, is confined to effective pixel areas so that the computational efficiency is improved. Firstly, the cause of invalid pixel areas is pointed out, and the necessity of adding masks is illustrated via quantitative analysis of the computing proportion of those areas in every area computing step. In particular, it is indicated that the larger the width-height ratio of the image is, the higher the speedup ratio benefiting from masks gets. After that, the architecture of ASIFT involving masks is given, and then a kind of data structure of mask, which is beneficial to low memory cost and high access efficiency, is put forward, and an analytical method for getting masks offline is also given. Finally, the experiment verifies that the efficiency of area computing part of ASIFT can be improved effectively by adding masks, and naturally the whole computing efficiency of ASIFT is improved to a certain extent, however, the margin of the improvement is limited by ideal area-speedup-ratio.

Keywords ASIFT algorithm; fully affine invariant; feature extraction; accelerated algorithm; computational efficiency

收稿日期:2013-07-06;最终修改稿收到日期:2014-11-21. 本课题得到国家自然科学基金(60804063,61175091)、江苏省自然科学基金(BK2010403)、图像信息处理与智能控制教育部重点实验室开放基金(200902)、东南大学优秀青年教师教学科研资助计划资助项目(3208001203)、东南大学创新基金(3208000501)、江苏省青蓝工程资助计划(1108000134)资助. 朱 博,男,1981年生,博士研究生,主要研究方向为定性空间关系推理、机器人交互与导航、环境感知等. E-mail: zhuboseu@163.com. 戴先中,男,1954年生,博士,教授,主要研究领域为复杂控制理论、机器人控制等. 李新德,男,1975年生,博士,副教授,主要研究方向为机器感知、信息融合、不确定推理和机器视觉等. 杨 伟,男,1989年生,硕士研究生,主要研究方向为机器视觉.

1 引言

近年来,图像处理和计算机视觉领域涌现出大量特征检测子(detector)和局部特征描述子(descriptor),相应局部特征被广泛用于图像检索、物体识别、机器人定位和宽基线匹配等研究^[1]。其中,具有仿射不变性的典型检测子有 MSER^[2]、Harris-Affine^[3]、Hessian-Affine^[4]等。Mikolajczyk 等人在文献[4]中对 6 种应用广泛的仿射不变性检测子进行了充分比较,参与比较的检测子包括: Harris-Affine、Hessian-Affine、MSER、EBR、IBR、Salient Regions,比较过程中使用的测试图像集含有结构场景和纹理场景,并且包含多种类型的变化(视点变化、尺度变化、光照变化、模糊、JPEG 压缩)。作者从重复度(repeatability)和区分性(distinctiveness)两方面对检测子比较后指出:在各种场景类型和变化类型下并不存在某种检测子性能最优;在许多情况下 MSER 具有更好的性能,其次是 Hessian-Affine。MSER 虽然性能突出,但并非完全尺度不变^[5],而 Hessian-Affine 尽管尺度不变,但是计算速度慢^[6]限制了其应用场合。事实上,Hessian-Affine 也并非完全仿射不变,文献[7]从平面对象的数字图像获取模型角度阐述了相应规范化方法(normalization methods)仿射不变性能受限的原因:模型中模糊算子同仿射变换不满足交换律,使得规范化过程不能精确消除仿射变换。另外,两类不考虑完全仿射不变性但被广泛使用的典型检测子是 SIFT 检测子(SIFT detector)^[7-8]和 Fast-Hessian 检测子^[9]。它们的一个共同特点是对图像斑点(blob)检测运算进行了适当近似(如对 LoG、Hessian 阵的行列式近似),在保证检测性能的基础上,使得运算速度大幅度提升,从而使得实时应用成为可能。

在局部特征描述子方面,众多性能各异的描述子亦被提出,常见的如形状上下文(shape context)^[10]、SIFT^[8]、PCA-SIFT^[11]、GLOH^[12]和 SURF^[9]等。Mikolajczyk 等人在文献[12]中对 2005 年之前出现的 10 种典型描述子进行了比较,指出在大多数测试中 GLOH 的性能最好,其次是 SIFT 描述子。然而,GLOH 的计算花费较大^[9],这限制了它在实时场合的应用,而 SIFT 描述子至今仍是最常用的描述子之一。Bay 等人^[9]在 2006 年提出 SURF 描述子,从文献[9]中的代表性实验结果看出,它的性能优于 SIFT、GLOH 和 PCA-SIFT 描述子并且计算代价

更低。

可以看出,在图像局部特征研究中,一方面,人们希望特征检测子和描述子具有更好的性能;另一方面,面对有限的计算资源,人们期望位于高层应用前端的特征提取算法简洁、高效,算法速度(效率)甚至成为其能否被广泛应用的必要条件。另外,由于并不存在一种特征提取算法性能最优且能够覆盖所有类型图像区域,因此应用中逐渐流行将多种互补的特征提取算法复合使用,例如文献[13]在场所语义建模研究中使用了 3 种互补特征。这种复合使用方式对单一特征提取速度提出了更苛刻的要求。因此,对于新提出的特征提取算法,其运算速度问题必须予以充分考虑。

ASIFT 特征提取算法是 Morel 和 Yu^[7]于 2009 年提出的一种完全仿射不变特征提取方法。SIFT 算法本质上对尺度参数进行模拟(simulation)而对旋转和平移参数进行规范化(normalization),ASIFT 算法在此基础上对仿射变换的另外两个参数——经、纬角进行了合理模拟,使得该算法具有完全仿射不变性。由文献[7]中实验可知,当待匹配图像对的转移倾斜指数(transition tilt)高达 36 时,匹配过程仍能获得 ASIFT 匹配点对,而此时 SIFT、Harris-affine、Hessian-affine 和 MSER 特征匹配均已失效。文献[7]还从数学上证明了 ASIFT 算法的完全仿射不变性。ASIFT 算法在提出之后已被用于多种场合,如文献[14]在人体行为识别研究中使用 ASIFT 算法,实现局部静态特征(local static features)提取;文献[15]将 ASIFT 算法用于人脸识别领域,使得在红外光谱范围内能够获得更好的匹配效果。ASIFT 算法尽管具有完美的仿射不变性能,但是低运算效率似乎成为其进一步广泛使用的障碍。网站 http://www.ipol.im/pub/demo/my_affine_sift/提供了 ASIFT 算法的在线测试实例,使用 Adam 测试图像对时,特征提取时间为 3 s(计算服务器为 Dell R910,含 4 个 8 核 Xeon 7560 2.3 GHz CPU^①),这说明即使以高性能服务器作为计算载体,现有 ASIFT 算法仍难以满足实时性要求,需要改进算法效率。为提高计算效率,文献[7]大致给出 3 种观点:(1)采用快速 SIFT 算法(如 PCA-SIFT 等);(2)各个 SIFT 处理过程并行运行;(3)采用两分辨率法。观点(1)和(2)均可以直接得到加速的 ASIFT 算法,

① 与 Guoshen Yu 和 Nicolas Limare 通信得知,在此对提供帮助的两位学者表示感谢。

而观点(3)虽然有助于大幅度减少算法复杂性,但确定二次采样因子和 M 个最优仿射变换的原则还有待探讨,不合适的取值将直接影响 ASIFT 特征提取和匹配效果. 本文发现影响 ASIFT 算法速度的另一因素是 ASIFT 算法在模拟摄像机经、纬角参数时会引入无效像素区域,它们在计算中将占有一定比例,影响运算效率. 在此基础上,本文提出使用“遮罩”来避免对无效区域的计算,并给出“遮罩”数据结构及其获取方式,从而在增加少量存储空间的前提下提高算法效率.

本文在第 2 节对 ASIFT 算法原理进行简要介绍并给出本文使用的一些术语定义;在第 3 节分析 ASIFT 算法引起无效像素区域的根源,定量分析指出使用“遮罩”的必要性并给出带有“遮罩”的 ASIFT 特征提取流程;在第 4 节提出“遮罩”的一种低存储空间、高访问效率的数据结构,并给出一种生成“遮罩”的解析方法;第 5 节中,对本文提出的方法进行实验验证,对实验现象进行必要分析和讨论;第 6 节对全文进行总结.

2 相关记号、定义及 ASIFT 算法原理

为方便读者参考,本节所用符号与文献[7]基本一致,算法细节及数学证明详见文献[7].

2.1 相关记号及定义

记号 1. x 方向上一维高斯算子^[7],记为

$$G_{\sigma}^x(x, y) = \frac{1}{\sqrt{2\pi}c\sigma} e^{-\frac{x^2}{2(c\sigma)^2}}$$

其中: σ 为方差, c 为常数,通常取 $c=0.8$ 以保证较好的抗锯齿效果,同理,有 y 方向上一维高斯算子 G_{σ}^y .

记号 2. x 方向一维卷积算子^[7],记为 $*_x$.

同理,有 y 方向一维卷积算子 $*_y$.

定义 1. 倾斜因子(tilt factor) t ^[7],定义为

$$t \triangleq \frac{1}{\cos\theta}$$

θ 为相机光轴与平面法向量夹角,称为相机倾斜角(见图 1).

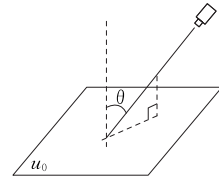


图 1 相机倾斜角示意

定义 2. 已知图像 $w(x, y)$, 定义 x 方向几何倾斜(geometric tilt)算子 T_t^x ^[7], 使得

$$T_t^x w(x, y) \triangleq w(tx, y).$$

同理,有 y 方向几何倾斜算子 T_t^y .

定义 3. 已知图像 $w(x, y)$, 定义 x 方向模拟倾斜(simulated tilt)算子 \mathbb{T}_t^x ^[7], 使得

$$\mathbb{T}_t^x w \triangleq T_t^x G_{\sqrt{t^2-1}} *_x w.$$

同理,有 y 方向模拟倾斜算子 \mathbb{T}_t^y .

定义 4. (R, \mathbb{T}_t^x) 空间: 以序偶 $\langle R(\varphi), \mathbb{T}_t^x \rangle$ 为元素构成的集合

$$\{ \langle R(\varphi), \mathbb{T}_t^x \rangle \mid \varphi \in [0, \pi), t > 1; \varphi = 0, t = 1 \},$$

称为 (R, \mathbb{T}_t^x) 空间, 其中 $R(\varphi)$ 为角度 φ 上的旋转变换. 该空间中某点作用于图像 $w(x, y)$ 时, 有 $\mathbb{T}_t^x R(\varphi) w(x, y)$.

定义 5. 某一图像处理算子 $P(\cdot)$, 如果其作用对象为图像 $w(x, y)$ 上一个或多个单连通域 $\mathcal{D} (\mathcal{D} \subseteq w(x, y))$, 则称其所作运算为区域运算, 同时称该算子为区域算子. 特别地, 如果 $P(\cdot)$ 的作用对象为整幅图像 ($\mathcal{D} = w(x, y)$), 则称对应运算为全区域运算, 简称全域运算.

定义 6. 某一图像处理算子 $P(\cdot)$, 如果其作用对象为图像上一组离散点, 则称其所作运算为点运算, 同时称该算子为点算子.

2.2 ASIFT 工作原理

如图 2 所示, 由 (R, \mathbb{T}_t^y) 空间某点作用于原始图像 u 得到图像 \bar{u} , 之后对图像 \bar{u} 提取 SIFT 特征并

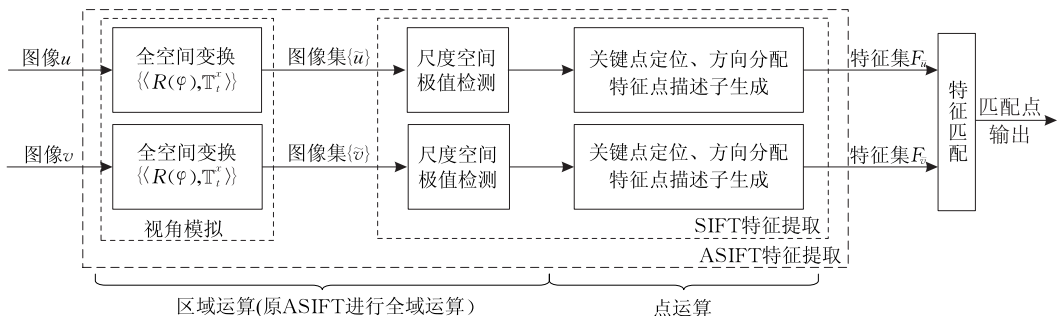


图 2 ASIFT 特征提取和匹配流程

将所得到的特征加入特征集 F_a , 如此遍历整个 (R, \mathbb{T}_t^y) 空间即得到完整的 ASIFT 特征集 F_a , 该特征集可用于特征匹配. 值得注意的是, 算法实现时为保证合理计算量, 通常对 (R, \mathbb{T}_t^x) 空间进行离散化处理. (R, \mathbb{T}_t^x) 空间变换也可形象地称为“视角模拟”.

整个 ASIFT 特征提取过程可以分为区域运算和点运算两部分(见图 2).

3 “遮罩”原理

文献[7]未细分 ASIFT 算法中区域算子的作用范围, 区域算子均进行全域运算(见图 2), 由作者给出的源代码也可以看出, 由于未细分作用范围, 代码笼统地将各区域算子作用于相应过程图像的整体. ASIFT 算法在进行 (R, \mathbb{T}_t^x) 全空间变换时会将大量无效像素引入临时图像, 它们以一定比例参与区域运算, 进而对运算效率造成影响. 本节首先分析无效像素区域产生的根源, 其次对其占有的比例进行量化分析, 指出为区域算子作用域增加“遮罩”的必要性, 最后给出带有“遮罩”的 ASIFT 特征提取流程. 由于连续模型的简便性, 本节仍沿用连续模型讨论相关问题.

3.1 无效像素区域产生原因

由 ASIFT 算法原理可知, 为了模拟仿射变换中经、纬角参数, ASIFT 算法首先要对原始图像 $w(x, y)$ 在整个 (R, \mathbb{T}_t^x) 空间变换, 而当进行某一变换 $(R(\varphi), \mathbb{T}_t^x)$ 时, 原始图像先要旋转 φ 角度. 我们知道, 通常二维图像为矩形, 且在计算机中以二维矩阵表示, 这样, 原始图像经旋转后必须规范为矩形图像以便以二维矩阵存储, 这使得在构成新矩形图像时不可避免地引入常值像素来填补空白区域. 所填补区域虽不影响特征提取结果, 但参与整个区域运算过程并占用运算时间, 故称其为无效像素区域. 旋转变换产生的无效像素区域实例见图 3 中灰色部分. (R, \mathbb{T}_t^x) 全空间变换会产生大量无效像素区域.

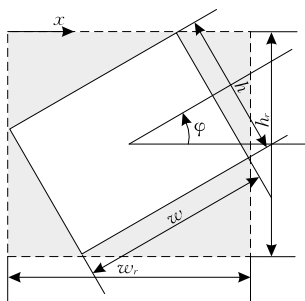


图 3 图像旋转后引入的无效像素区域(灰色部分)

3.2 使用“遮罩”的必要性分析

由于 ASIFT 算法在开始时便将大量无效像素区域引入临时图像, 导致后续所有进行全域运算的区域算子产生大量无效运算, 因此有必要在每步运算中使用与当前有效像素区域大小、姿态一致的“遮罩”, 将无效区域屏蔽, 从而提高计算效率. 下面, 定量分析由原 ASIFT 算法中进行全域运算的 4 类区域算子所处理的总无效像素区域同总作用区域的面积比, 从定量角度给出引入“遮罩”的必要性, 并给出理想区域加速比概念.

(1) 旋转变换算子 R . 由于几何变换通常采用逆映射方式求得新图像, 因此, 实际上从旋转变换开始便引起无效运算. 在 (R, \mathbb{T}_t^x) 全空间上分析, 且仅需 $\varphi \in [0, \pi), t > 1$; 设原始图像的宽、高分别为 w, h , 旋转角度 φ 后, 图像宽、高分别为 w_r, h_r ; 不妨设 x 轴方向水平向右(见图 3); 设在 (R, \mathbb{T}_t^x) 空间上共进行 n 次 \mathbb{T}_t^x , 事实上由于使用连续模型, 所以有 $n \rightarrow \infty$. 这样, (R, \mathbb{T}_t^x) 空间上 R 涉及的无效像素区域面积总和为

$$\begin{aligned} \hat{S}_R &= n \cdot 2 \int_{\varphi=0}^{\pi/2} \left(2 \cdot \frac{1}{2} h \cos \varphi \cdot h \sin \varphi + \right. \\ &\quad \left. 2 \cdot \frac{1}{2} w \sin \varphi \cdot w \cos \varphi \right) d\varphi \\ &= (h^2 + w^2) n. \end{aligned}$$

R 的总作用区域面积为

$$\begin{aligned} S_R &= n \cdot 2 \int_{\varphi=0}^{\pi/2} h_r w_r d\varphi \\ &= n \cdot 2 \int_{\varphi=0}^{\pi/2} (h \cos \varphi + w \sin \varphi)(h \sin \varphi + w \cos \varphi) d\varphi \\ &= n \cdot [(h^2 + w^2) + h w \pi]. \end{aligned}$$

两者比值为

$$\begin{aligned} \frac{\hat{S}_R}{S_R} &= \lim_{n \rightarrow \infty} \frac{(h^2 + w^2) n}{[(h^2 + w^2) + h w \pi] n} \\ &= \frac{h^2 + w^2}{h^2 + w^2 + h w \pi}. \end{aligned}$$

上述比值在 $h = w$ 时取得最小值:

$$\min \frac{\hat{S}_R}{S_R} = \frac{2}{2 + \pi} \approx 38.898\%.$$

(2) \mathbb{T}_t^x 中一维高斯模糊算子 $G_{\sqrt{t^2-1}}$. 一维高斯模糊算子作用域与旋转变换 R 完全一致, 所以易得

$$\frac{\hat{S}_G}{S_G} = \frac{h^2 + w^2}{h^2 + w^2 + h w \pi}.$$

同样, 在 $h = w$ 时取得最小值:

$$\min \frac{\hat{S}_G}{S_G} = \frac{2}{2 + \pi} \approx 38.898\%.$$

(3) \mathbb{T}_t^x 中几何倾斜算子 T_t^x . 通常采用逆映射方式实现几何倾斜运算, 因此在倾斜后的图像上计算

相关面积比. 设沿 x 轴方向作倾斜变换后图像宽为 w_{rt} , 如图 4 所示.

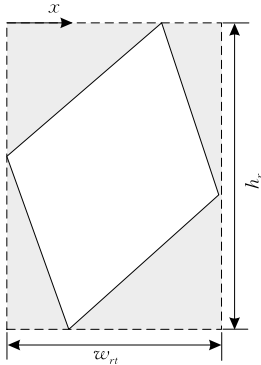


图 4 x 轴方向倾斜变换

(R, T_t^x) 空间上 T_t^x 涉及的无效像素区域面积为

$$\hat{S}_t = 2 \int_{t=1}^{\infty} \int_{\varphi=0}^{\pi/2} \left(2 \cdot \frac{1}{2} h \cos \varphi \cdot \frac{h \sin \varphi}{t} + 2 \cdot \frac{1}{2} w \sin \varphi \cdot \frac{w \cos \varphi}{t} \right) d\varphi dt = (h^2 + w^2) \ln |t|_1^{\infty}.$$

T_t^x 的总作用区域面积为

$$S_t = 2 \int_{t=1}^{\infty} \int_{\varphi=0}^{\pi/2} (h \cos \varphi + w \sin \varphi) \left(\frac{h \sin \varphi}{t} + \frac{w \cos \varphi}{t} \right) d\varphi dt = (h^2 + w^2) \ln |t|_1^{\infty} + h w \cdot 2 \int_0^{\pi/2} 1 d\varphi \ln |t|_1^{\infty} = (h^2 + w^2) \ln |t|_1^{\infty} + h w \pi \ln |t|_1^{\infty}.$$

两者比值为

$$\frac{\hat{S}_t}{S_t} = \frac{(h^2 + w^2) \ln |t|_1^{\infty}}{(h^2 + w^2) \ln |t|_1^{\infty} + h w \pi \ln |t|_1^{\infty}} = \lim_{t \rightarrow \infty} \frac{(h^2 + w^2) \ln |t|}{(h^2 + w^2) \ln |t| + h w \pi \ln |t|} = \frac{h^2 + w^2}{h^2 + w^2 + h w \pi}.$$

上述比值在 $h = w$ 时取得最小值:

$$\min \frac{\hat{S}_t}{S_t} = \frac{2}{2 + \pi} \approx 38.898\%.$$

(4) SIFT 的尺度空间极值检测算子. 极值检测算子作用域与几何倾斜算子 T_t^x 完全一致, 所以易得

$$\frac{\hat{S}_{\text{SIFT-E}}}{S_{\text{SIFT-E}}} = \frac{h^2 + w^2}{h^2 + w^2 + h w \pi}.$$

同样, 在 $h = w$ 时取得最小值:

$$\min \frac{\hat{S}_{\text{SIFT-E}}}{S_{\text{SIFT-E}}} = \frac{2}{2 + \pi} \approx 38.898\%.$$

后续图像处理过程对关键点进行点运算, 不再受无效区域影响. 由上述 4 步定量分析可以看出, 原 ASIFT 特征提取过程中各区域算子处理的总无效像素区域同总作用区域面积比均为 $(h^2 + w^2) / (h^2 + w^2 + h w \pi)$. 令

$$\lambda = \frac{k^2 + 1}{k^2 + \pi k + 1} (k = w : h),$$

称其为原 ASIFT 算法的理想区域加速比公式, 其值为理想区域加速比, 可见 λ 仅与被处理图像(或被处理图像区域)宽高比有关. 当被处理图像为正方形时总无效像素区域所占比例最小, 达 38.898%; 当被处理图像为一般矩形时理想区域加速比较大, 典型地, 当 $k = 4 : 3$ 时, $\lambda = 39.873\%$; $k = 16 : 9$ 时, $\lambda = 42.691\%$. 易得 λ 为 k 的增函数, 由此可知, 被处理区域宽高比越大, 无效像素区域所占比例越大, 在其上耗费的运算时间比例也越大.

通常假设区域算子运算时间同作用区域大小成正比, 则当无效区域被屏蔽时, 区域算子运算速度理想情况下将提升 λ . 从上述定量分析可知, 使用“遮罩”对无效区域进行屏蔽有其必要性, 特别地, 被处理区域宽高比越大, 通过“遮罩”屏蔽无效像素区域获得的加速比越高.

3.3 带有“遮罩”的 ASIFT 特征提取流程

带有“遮罩”的 ASIFT 特征提取流程图如图 5 所示. 其中, 在定量分析述及的 4 个区域运算环节处

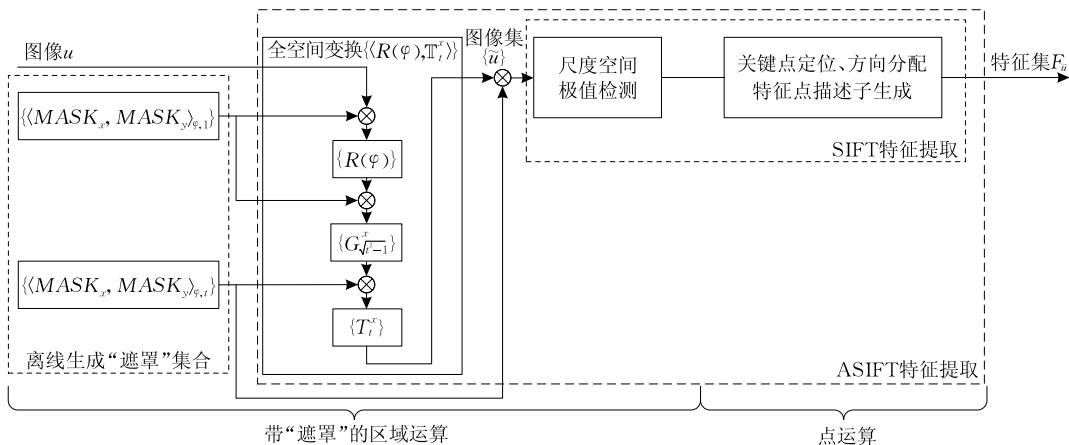


图 5 带有“遮罩”的 ASIFT 特征提取流程

使用“遮罩”,当进行每一区域运算时,从“遮罩”集合中取出与当前图像有效像素区域大小、姿态一致的“遮罩”,以屏蔽无效像素区域,从而提高计算效率; $\{\langle MASK_x, MASK_y \rangle_{\varphi,1}\}$ 和 $\{\langle MASK_x, MASK_y \rangle_{\varphi,t}\}$ 为离线生成的两种“遮罩”集合,将于下一节详细介绍。

4 “遮罩”的数据结构与生成方法

理论算法的计算机实现需要保持合理计算量和一定精度,因此需要对连续 (R, \mathbb{T}_t^r) 空间进行合理离散化。文献[7]指出通常需要根据实验确定采样步长,并给出了一般离散化原则:设倾斜采样步长为 Δt ,旋转变换采样步长为 $\Delta\varphi$,则有 $t = (\Delta t)^n$, $n = 0, 1, 2, \dots$ 和 $\varphi = m\Delta\varphi$, $m = 0, 1, 2, \dots$,通常取 $\Delta t = \sqrt{2}$, $\Delta\varphi = 72^\circ/t$ 。当进行 ASIFT 特征提取时,原始图像根据 Δt 和 $\Delta\varphi$ 在 (R, \mathbb{T}_t^r) 空间上变换,变换后每幅图像再由 SIFT 过程处理。由于每一区域运算步骤对应一个“遮罩”,因此“遮罩”与相应图像一样是 t 和 φ 的函数,不同之处在于“遮罩”可以离线生成,进而节省在线处理时间。

直观上,可以以一幅二值图像来表示一个“遮罩”,其中不妨以 1 表示该位像素有效,当有区域需要屏蔽时程序访问该二值图像,仅对其上“1”对应的当前图像像素进行处理。事实上,这种数据结构不仅引入大量“遮罩”图像,造成内存耗费,而且使得各区域算子运算时遍历两幅图像,计算效率低下。由于“遮罩”为单连通域,因此它可由边界唯一确定,这样,以边界“线信息”取代“面信息”可减少存储空间并能避免遍历二维图像,从而提高访问效率。本文采用边界数据对“遮罩”进行描述。

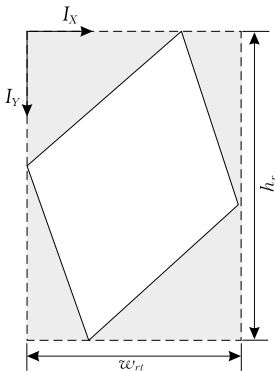


图 6 “遮罩”实例

图 6 中给出一个“遮罩”实例,它与图 4 所示过程图像对应,“遮罩”内部区域参与区域运算,而外部

区域被屏蔽。由于 ASIFT 算法实现过程经常涉及一维运算,为使用方便,对同一“遮罩”采用两种描述方式: x 方向遮罩、 y 方向遮罩。两种“遮罩”均以 $n_{(\cdot)} \times 2$ 维矩阵表示,分别形式化如下:

$$MASK_x = \begin{bmatrix} \vdots & \vdots \\ I_{i,x1} & I_{i,x2} \\ \vdots & \vdots \end{bmatrix}_{n_y \times 2}, I_{i,x1} \leq I_{i,x2};$$

$$MASK_y = \begin{bmatrix} \vdots & \vdots \\ I_{i,y1} & I_{i,y2} \\ \vdots & \vdots \end{bmatrix}_{n_x \times 2}, I_{i,y1} \leq I_{i,y2}.$$

其中, $I_{(\dots)} \geq 0$ 表示二维数字图像平面上坐标, $n_{(\cdot)}$ 表示图像 $y(x)$ 方向分辨率, $MASK_x$ 的第 i 行为 y 轴坐标 $I_y = i - 1$ 时相应“遮罩”边界的 x 坐标 $I_{i,x1}$ 、 $I_{i,x2}$,同样, $MASK_y$ 的第 i 行为 x 轴坐标 $I_x = i - 1$ 时相应“遮罩”边界的 y 坐标 $I_{i,y1}$ 、 $I_{i,y2}$ 。 (R, \mathbb{T}_t^r) 空间上的“遮罩”集合记为 $\{\langle MASK_x, MASK_y \rangle_{\varphi,t}\}$ 。

下面,给出一种生成“遮罩”边界的解析方法:

$\varphi = 0, \pi/2$ 时,无需增加“遮罩”,因此只考虑区间 $\varphi \in (0, \pi/2)$ 和 $\varphi \in (\pi/2, \pi)$,设取整算子为 $Z(\cdot)$,图像为 $w \times h$ 像素,图像坐标系如图 6 所示,解直线方程易得 $\varphi \in (0, \pi/2)$ 时,有

$$I_{x1} = \begin{cases} Z\left(-\frac{1}{t} \cdot \cot\varphi \cdot [(i-1) - w\sin\varphi]\right) \\ i-1 \in [0, w\sin\varphi] \\ Z\left(\frac{1}{t} \cdot \tan\varphi \cdot [(i-1) - w\sin\varphi]\right) \\ i-1 \in [w\sin\varphi, h\cos\varphi + w\sin\varphi] \end{cases},$$

$$I_{x2} = \begin{cases} Z\left(\frac{1}{t} \cdot \tan\varphi \cdot (i-1) + \frac{w\cos\varphi}{t}\right) \\ i-1 \in [0, h\cos\varphi] \\ Z\left(-\frac{1}{t} \cdot \cot\varphi \cdot (i-1) + \frac{h}{t\sin\varphi} + \frac{w\cos\varphi}{t}\right) \\ i-1 \in [h\cos\varphi, h\cos\varphi + w\sin\varphi] \end{cases},$$

当 $\varphi \in (\pi/2, \pi)$ 时,有

$$I_{x1} = \begin{cases} Z\left(\frac{1}{t} \cdot \tan\varphi \cdot [(i-1) + h\cos\varphi]\right) \\ i-1 \in [0, -h\cos\varphi] \\ Z\left(-\frac{1}{t} \cdot \cot\varphi \cdot [(i-1) + h\cos\varphi]\right) \\ i-1 \in [-h\cos\varphi, w\sin\varphi - h\cos\varphi] \end{cases},$$

$$I_{x2} = \begin{cases} Z\left(-\frac{1}{t} \cdot \cot\varphi \cdot (i-1) + \frac{h\sin\varphi}{t}\right) \\ i-1 \in [0, w\sin\varphi] \\ Z\left(\frac{1}{t} \cdot \tan\varphi \cdot (i-1) + \frac{h\sin\varphi}{t} - \frac{w}{t\cos\varphi}\right) \\ i-1 \in [w\sin\varphi, w\sin\varphi - h\cos\varphi] \end{cases},$$

同理, I_{y_1} 和 I_{y_2} 的解析表达为上述表达式的反函数, 不再赘述. 值得注意的是, 由于 SIFT 特征提取过程要使用图像金字塔, 因此需要在相应步对“遮罩”进行半采样——取“遮罩”奇数行并将坐标值减半取整; 另外, 旋转变换算子 R 和 T_t^x 中一维高斯模糊算子 $G^x_{\sqrt{t^2-1}}$ 计算时图像未进行几何倾斜变换, 但仍可采用上述解析表达式作为“遮罩”的生成公式, 此时只需令 $t=1$, “遮罩”集合记为 $\{\langle MASK_x, MASK_y \rangle_{\varphi, 1}\}$.

5 实验与讨论

本节以实验验证前述算法的有效性. 实验硬件环境: CPU 为 Intel Core2 P8400 (双核, 2.26 GHz), 关闭 Intel SpeedStep 模式; 芯片组为 Intel GM45; 使用 DDR3 内存 (容量 4 GB, 主频 1066 MHz). 实验软件环境: 操作系统为 Windows 7 专业版; 编译器为 Intel C++ Compiler Professional 11.1.

为避免同一算法的不同代码实现所造成的运算速度差异, 本文在文献[7]提供的源代码 (下载地址: http://www.ipol.im/pub/alg/my_affine_sift/) 基础上增添“遮罩”, 不对与“遮罩”无关的部分进行任何修改, 并且不对增加“遮罩”的代码段中的原数据结构及算法做任何代码级优化. 原代码中大部分运行参数保持不变, 而个别修改的参数在后文给出进一步说明. 使用 Adam 图像对作为测试图像, 将原始图像规范到 400×300 分辨率. 实验中, 对原 ASIFT 算法和带“遮罩”的 ASIFT 算法 (Mask-ASIFT) 的特征提取效果进行了对比 (见表 1); 在无“遮罩”和有“遮罩”两种情况下, 对前述受无效像素区域影响的 4 类区域算子进行计时比较, 并对增加“遮罩”后的提速效果进行了分析; 为说明提速效果的可重复性, 在同一运行条件下代码被重复运行和计时 21 次; 最后, 对整个 ASIFT 特征提取过程的速度提升效果也进行了实验说明.

表 1 原 ASIFT 和 Mask-ASIFT 特征提取效果对比

项目	图像 1 特征点数	图像 2 特征点数
原 ASIFT	3003	2598
Mask-ASIFT	3000	2599
Mask-ASIFT (SIFT 极值检测时边界 收缩 2 像素)	2977	2587

如表 1 所示, Mask-ASIFT 同原 ASIFT 算法提取的特征点数目基本相同, 差异特征点出现于有效

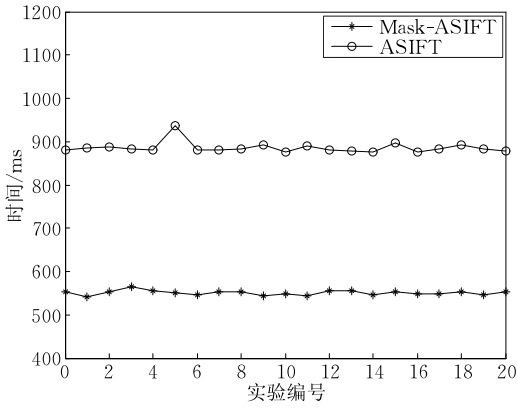
像素区域的边界附近. 原因在于: 原 ASIFT 算法中有效像素区域边界位于图像内部, SIFT 过程在其附近只需按常规做卷积等运算; 而 Mask-ASIFT 算法中有效像素区域的边界是实际处理的图像物理边界, 为进行卷积等运算需要在边界外侧增加有限宽像素带, 使得边界响应同原 ASIFT 算法不同, 造成边界附近出现差异特征点.

图 7 各子图横轴为代码运行次数编号, 共进行 21 次实验, 纵轴表示运行时间. 当图像分辨率被规范到 400×300 时, 根据理想区域加速比公式可知此时理想区域加速比为 39.873%. 旋转变换算子 R (图 7(a))、一维高斯模糊 $G^x_{\sqrt{t^2-1}}$ (图 7(b)) 和几何倾斜算子 T_t^x (图 7(c)) 的实验加速比平均值分别为 37.812%、31.232% 和 31.079%, 均略低于理想区域加速比. 事实上, 算法实现时对 (R, T_t^x) 空间进行了离散化处理, 且由于实际代码需要进行变量初始化、内存空间开辟等额外处理, 使得区域运算时间同区域面积的正比关系并不严格成立, 导致出现实际加速比低于理想区域加速比, 今后可通过代码级优化或增加并发度使之接近甚至超过理想区域加速比. 图 7(d) 中, SIFT 尺度空间极值检测平均加速比为 50.422%, 这或许得益于某些代码的编写, 使得编译器优化生成的机器码能够充分利用 CPU 向量化 (Vectorization) 或其它特性, 增加了代码并发度, 造成实际加速比高于理想区域加速比. 但因受制于理想区域加速比和 CPU 并行化程度, 实际加速比并不会高出理想区域加速比很多.

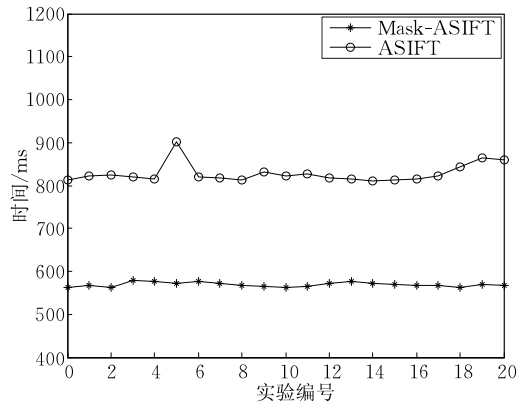
理想情况下, “遮罩”的使用不对后续点运算产生影响, 而事实上, 如前所述, Mask-ASIFT 算法中有效像素区域的边界是实际处理的图像物理边界, 在边界附近出现强响应, 使得经极值检测后边界附近存在大量关键点, 这些关键点均参与点运算, 从而导致后续点运算量增加、运算时间加长 (图 8 三角线条所示). 为解决该问题, 可适当缩小极值检测区域. 当极值检测边界缩小 2 像素时, Mask-ASIFT 算法获得的特征点数目略有减小 (见表 1 末行), 而点运算速度比原 ASIFT 算法略有提高 (见图 8 星号线条所示).

理论上, “遮罩”的增加只能提高区域相关运算的效率, 而通常我们更关注整个特征提取过程的效率, 图 9 给出“遮罩”增加前后整个 ASIFT 特征提取时间对比.

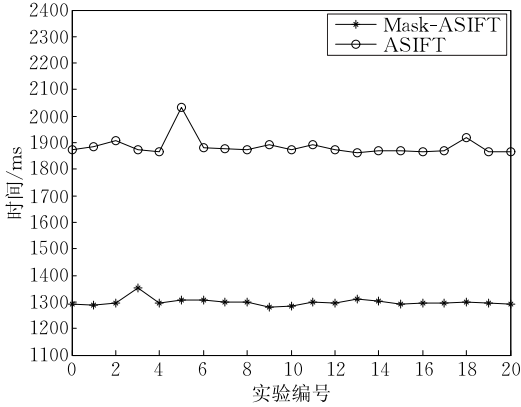
图 9(a) 为极值检测边界未经修改时的耗时对比, “遮罩”的增加能够使运算效率平均提升 25.908%; 图 9(b) 为极值检测边界缩小 2 像素时的



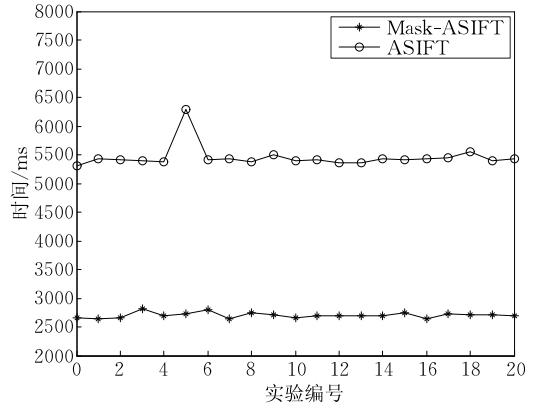
(a) ASIFT中全部旋转变换时间(pic1)



(b) ASIFT中全部一维高斯模糊时间(pic1)



(c) ASIFT中全部几何倾斜运算时间(pic1)



(d) ASIFT中全部SIFT所含区域运算总时间(pic1)

图 7 增加“遮罩”前后 4 类区域算子耗时对比

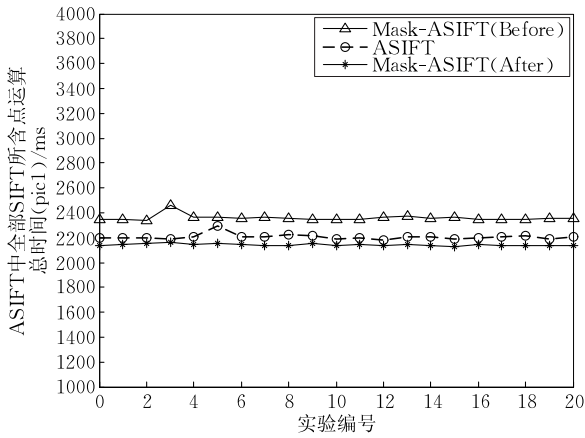
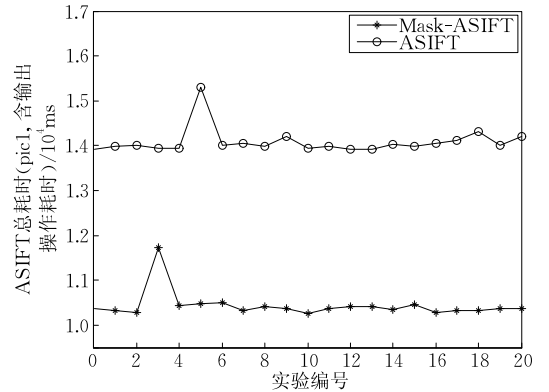
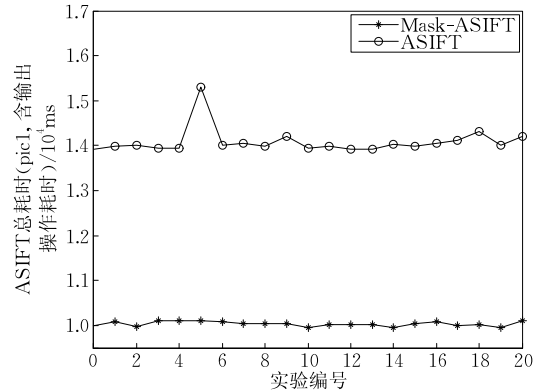


图 8 “遮罩”对点运算影响

耗时对比,“遮罩”的增加能够使运算效率平均提升 28.773%。ASIFT 算法整体运算速度不仅取决于区域运算部分,而且取决于点运算部分,而经验上,通常点运算量小于区域运算量(实例参见图 7 和图 8,点运算占 2.2s,区域运算占 9.1s),因此对区域运算部分的速度提升显得更加重要。ASIFT 算法中点运算部分的计算量取决于图像内容并与应用场合有关,采用高效关键点描述算法(如 SURF 等)可进一步使之降低。



(a) 未修改极值检测边界的 ASIFT 总耗时(pic1, 含输出操作)



(b) 缩小极值检测边界向的 ASIFT 总耗时(pic1, 含输出操作)

图 9 “遮罩”增加前后 ASIFT 特征提取时间对比

6 结束语

ASIFT 算法是一种经过数学严格证明的完全仿射不变特征提取算法,本文研究了原 ASIFT 算法忽略的一个方面,指出原 ASIFT 算法在初始时刻会引入大量无效像素区域,如果不区分区域算子作用范围,无效像素区域将参与整个 ASIFT 算法的区域运算过程,导致计算资源浪费、运算效率降低.本文提出使用“遮罩”来屏蔽无效像素区域,定量分析了使用“遮罩”的必要性,特别指出 ASIFT 算法处理图像的宽高比越大,经“遮罩”处理后获得的加速比越高;给出“遮罩”实现时所使用的一种高效数据结构,同时给出离线获取“遮罩”的一种解析方法.在标准图像对上进行了实验,验证了“遮罩”对 ASIFT 算法中区域运算部分及整体的速度提升效果,并指出增加“遮罩”后要适当缩小 SIFT 极值检测的区域,减少不必要点运算.本文只讨论了提高 ASIFT 算法效率的一个侧面,其由 ASIFT 算法自身特性引起,事实上,一些相关理论能够进一步提高 ASIFT 算法中区域算子本身和点运算本身的计算效率,多种提速方法的综合应用将有利于实时实现 ASIFT 算法.另外,一些并行技术的使用也将有助于实时实现 ASIFT 算法,这些均为今后需要研究的方向.

参 考 文 献

- [1] Li Jing, Allinson N M. A comprehensive review of current local features for computer vision. *Neurocomputing*, 2008, 71(10-12): 1771-1787
- [2] Matas J, Chum O, Urban M, Pajdla T. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 2004, 22(10): 761-767
- [3] Mikolajczyk K, Schmid C. Scale & affine invariant interest point detectors. *International Journal on Computer Vision*, 2004, 60(1): 63-86
- [4] Mikolajczyk K, Tuytelaars T, Schmid C, et al. A comparison of affine region detectors. *International Journal of Computer Vision*, 2005, 65(1): 43-72
- [5] Forsen P E, Lowe D G. Shape descriptors for maximally stable extremal regions//*Proceedings of the 11th IEEE International Conference on Computer Vision*. Rio de Janeiro, Brazil, 2007: 1-8
- [6] Loncomilla P, Ruiz-del-Solar J. Robust object recognition using wide baseline matching for robocup applications//*Ubbo V et al, eds. RoboCup 2007: Robot Soccer World Cup XI*. Berlin/Heidelberg, GER: Springer, 2008: 441-448
- [7] Morel Jean-Michel, Yu Guoshen. ASIFT: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences*, 2009, 2(2): 438-469
- [8] Lowe D. Distinctive image features from scale-invariant key points. *International Journal of Computer Vision*, 2004, 60(2): 91-110
- [9] Bay H, Tuytelaars T, Van Gool L. Surf: Speeded up robust features//*Proceedings of the 9th European Conference on Computer Vision*. Graz, Austria, 2006: 404-417
- [10] Belongie S, Malik J, Puzicha J. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, 2(4): 509-522
- [11] Ke Y, Sukthankar R. PCA-SIFT: A more distinctive representation for local image descriptors//*Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Pittsburgh, USA, 2004: 511-517
- [12] Mikolajczyk K, Schmid C. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, 27(10): 1615-1630
- [13] Ranganathan A, Dellaert F. Semantic modeling of places using objects//*Proceedings of the 2007 Robotics: Science and Systems Conference*. Atlanta, USA, 2007: 27-30
- [14] Minhas R, Baradarani A, Seifzadeh S, Jonathan Wu Q M. Human action recognition using extreme learning machine based on visual vocabularies. *Neurocomputing*, 2010, 73(10-12): 1906-1917
- [15] Boisier B, Billiot B, Abdesslem Z, Gouton P, Hardeberg J Y. Extraction and fusion of spectral parameters for face recognition//*Proceedings of the International Society for Optical Engineering*. San Francisco, USA, 2011: 1-6



ZHU Bo, born in 1981, Ph.D. candidate. His research interests include qualitative spatial-relationship reasoning, robot interaction and navigation, environment perception, etc.

DAI Xian-Zhong, born in 1954, Ph.D., professor. His research interests include complex control theory, robot control, etc.

LI Xin-De, born in 1975, Ph.D., associate professor. His research interests include machine perception, information fusion, uncertain reasoning, computer vision, etc.

YANG Wei, born in 1989, M.S. candidate. His research interest is computer vision.

Background

In recent years, a lot of valuable achievements, which are beneficial to many related research areas, such as image retrieval, object recognition, robot localization, and wide baseline matching, have been obtained in local image feature research. One of the interesting points of this research is to find a kind of local feature which is fully affine invariant. The ASIFT algorithm invented by Morel and Yu keeps scale invariant and truly affine invariant, and its fully affine invariant attribute has been proved mathematically. However, the low computational efficiency becomes one of the limitations that restrict the algorithm's broad application. The authors of ASIFT have given three ways to improve its computational efficiency: (1) adopt the accelerated version of SIFT; (2) implement the SIFT subroutines in parallel; (3) use two-resolution method. Apart from these methods, a novel way is introduced to improve the algorithm's efficiency in this paper. In our work, masks are used to shield invalid pixel areas which are caused by view simulation, thus the area computing part, which is more time-consuming in ASIFT, is confined to effective pixel areas so that the computational efficiency can be improved. Ideally, the efficiency of area computing part can be improved by ideal area-speedup-ratio.

The main advantage of our method called Mask-ASIFT is that the speed of area computing part of ASIFT can be effectively improved without significantly increasing memory occupation and algorithm complexity. Furthermore, many other accelerated methods, such as some accelerated versions of the operators which appear in area computing part or point computing part, can also be combined with Mask-ASIFT, so that the ASIFT feature can be abstracted in a more real-time manner.

This work was supported by the National Natural Science Foundation of China (Nos. 60804063 and 61175091), the Natural Science Foundation of Jiangsu Province (No. BK2010403), the Public Funds of Image Processing and Intelligent Control Key Laboratory of Chinese Education Ministry (No. 200902), the Excellent Young Teacher Foundation of Southeast University in Teaching and Science Research (No. 3208001203), the Science and Technology Innovation Foundation in Southeast University (No. 3208000501) and the Qing Lan Project of Jiangsu Province (No. 1108000134). Our group has been working on advanced algorithms of object recognition and environment perception, and their high efficient computing techniques.