

指导教师： 杨涛

提交时间： 2015/3/28

The task of
Digital Image Processing

数字图像处理

School of Computer Science

No: 1

姓名： 魏 洁

学号： 2012302408

班号： 10011201

保形半投影扭曲图像拼接

Che-Han Chang 莊永裕 国立台湾大学
佐藤洋一 东京大学

摘要

本文提出了一种新颖的参数化扭曲法，它是投影变换和相似变换的空间组合。基于分析投影变换的基础，在给出涉及两个输入图像的投影变换上，我们的方法顺利地将重叠区域转换为非重叠区域的投影变换，并且所得的扭曲逐渐从投影上变为在图像上的相似。所提出的这种扭曲法有着投影和相似扭曲两方面的优势。它为投影扭曲提供了良好的对准精度，同时为相似扭曲保持着各个图像的透视性。它也可以与更先进的基于局部扭曲的对准方法结合起来，例如尽可能投影扭曲，从而达到更好的对准精度。用所提出的扭曲法，视野可以通过以较少的投影失真（拉伸形状和放大尺寸）拼接图像来扩展。

1.引言

图像拼接是将一组图像结合成为一个场景有更广阔视野的更大的图像的过程。对于鲁棒性，图像拼接通常是通过发现带对准图像全局参数扭曲来解决。[13]受欢迎的全局扭曲有相似性，仿射和投影。不管其鲁棒性，对于所有的场景和动作，全局扭曲通常不够灵活。例如，投影扭曲只能为平面场景或者无视差相机运动提供精确对准。

为应对全局扭曲模型的不足并提高对准精度，近期，一些局部扭曲模型已经被提出，如平滑变化仿射(SVA)扭曲[9]和尽可能投影(APAP)扭曲[13]。为了获得更好的对准精度，而不是依赖于单一的全局扭曲，这些方法采用重叠区域的多个局部参数扭曲去解决错位/偏差。对于不重叠区域，投影(仿射)正规化被用于顺利推断超出图像重叠和类似于整体的全局变换。例如，APPA

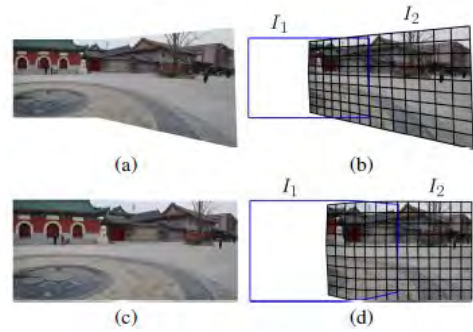


图 1(a,b)APAP 扭曲。(c,d) APAP+我们的翘曲。需要注意的是在 APAP(b)中四边形的大小不同而且它们的形状扭曲，显示了严重的尺寸/形状失真特别是在 I_2 的非重叠区域。另一方面，在图(d)中我们的扭曲保持形状并且规则化四边形的尺寸，显示出较少的失真。

扭曲在重叠的区域使用局部投影扭曲，同时使用移动 DLT 将顺利推断局部投影扭曲成非重叠区域。因此，全局投影允许局部偏差导致模型的不足[13]。然而，因为 APAP 扭曲试图全局性地类似投影扭曲，它面临着与投影扭曲相同的问题：形状/区域的失真，即，拼接图像的一部分被严重地拉伸和非均匀的放大。

考虑到传统的两视图的拼接问题，在图 1，图像 I_1 和 I_2 被拼接，并且 I_1 作为基础图像。图 1(a)显示 APAP 扭曲的结果。APAP 扭曲局部适应不同的变换，从而可以在两幅图像的重叠区域得到更精准的对准。在重叠区域的扭曲被推断到非重叠区域得到的结果接近全局投影扭曲（图 1(b)）。从这个例子中看出，通过这种投影扭曲 I_2 中的非重叠区域在大小和形状上严重变形。当为了进一步扩充视野在 I_2 上拼接下一幅图像时，这种失真甚至可能会恶化。Hartley 和 Zisserman 指出另一个值得说明的是，在 I_2 上应用投影扭曲可以有效地改变 I_1 的角度。因此 I_2 是 I_1 的扭曲的扩展视图。拼接过的图像展示了 I_1 的一个更为广阔的视角，一

个单一的视角。圆柱和球形的扭曲通过提供多维视图解决角度扭曲视图相对有限的问题。遗憾的是，这些扭曲通常是曲线并且只有当所有图像在同一相机中心拍摄才会有效。

由于具有广阔视角的单一视角图像不可避免的引起了严重的形状/尺寸失真，必须提供一个多视角的拼接图像来解决。初步的想法是为在重叠区域获得更好的调整采用投影扭曲而在非重叠区域保持每个视图的角度采用相似扭曲。我们选择投影扭曲因为它是最灵活的全局扭曲。我们选择相似扭曲是因为它是经过平移、均匀缩放和旋转得到的，因此它不会引起形状畸变和非均匀缩放。从另一个角度来讲，相似扭曲可以解释为平移、缩放和摄像机平面旋转的一个组合，这使观察的方向不会改变，从而保持了角度。此外，要通过图像建立一个平滑的扭曲，这需要在经过投影扭曲的区域和经过相似扭曲的区域之间有一个平稳的变换。

本文提出了一种新颖的参数化扭曲法，*保形半投影扭曲*，它是投影变换和相似变换的空间组合。这样的组合需要同时考虑调整 and 角度保持的灵活性。通过图 1 中给出的 I_1 和 I_2 ，我们首先可以估算出他们之间的全局投影变换。从中，我们通过一种新的外推法确定了投影变换的相似变换。直观的说，我们的扭曲通过投影扭曲在重叠区域进行调整而通过相似扭曲保持了非重叠区域的图像角度。这种外推法也被用于在经过投影扭曲和相似扭曲的区域之间形成平稳的变换。我们也建议将我们所提出的扭曲与 APAP 扭曲进行整合。这种组合扭曲可以像 APAP 扭曲那样在局部的程度上对准图像，并且可以像所提出的扭曲那样生成多维视角。如图 1(c)(d)所示，采用组合扭曲拼接的图像具有多维视角同时保持着精确对准。通过所提出的扭曲，视野可以在尺寸、形状和线条上无严重失真地扩展。

2. 相关研究

Szeliski 关于图像拼接有一个全面的调

查[10]。传统的方法 [11, 1] 假设相机运动只包含旋转。图像拼接只在视野范围内进行。一种投影是为获得 2D 合成图像而勘察平面视野范围。少数以尽量减少由于投影引起的视觉失真为目的的投影模型被提出。Brown 等人在他们的自动拼接计划中采用了球面投影[1]。Zelnik-Manor 等人[14]用了一个多平面投影来代替圆柱投影。Kopf 等人[8]提出了全局圆柱并且局部透视的局部适应投影。Carroll 等人[2]为降低广角图像畸变提出了内容保持投影。当这些模型的基本假设并不符合时误差就会产生，这是可以用后处理方法（例如：去鬼影和混合）隐藏误差。

图像拼接技术通常采用参数变换进行全局或局部的图像对准。Gao 等人[6]为专门处理含有两个主要平面的场景提出了双单应性扭曲。这个翘曲函数由两个单应性与空间变化权重的线性组合所定义。由于他们的扭曲是基于投影变换的，因此所得到的图像会有投影失真（区域的伸展和扩大）。他们提出了一种后处理来减轻这种失真。Lin 等人[9]提出了一种全局并且允许局部变形的平滑变化仿射的领域拼接。Zaragoza 等人[13]为了更好的调整提出了全局投影并且允许局部误差的尽可能投影扭曲。我们的做法是在分析投影变换的基础上。Chum 等人[4]通过坐标变化推导出为计算投影变换的几何误差的公式。类似的，我们通过坐标变换来改述投影变换。Chum 和 Matas [3]分析了在投影变换下的局部比例变化应将其用于仿射纠正。我们遵循了同样的分析来设计我们的扭曲。

3. 扭曲

本节描述了翘曲函数 $w : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ 从 (x, y) 到 (x', y') 之间的映射。给定两个图像空间上的投影变换 \hat{H} ，我们首先用一条线将 \mathbb{R}^2 分成两 R_H 和 R_L 。对于 $(x, y) \in R_H$ 令 $w(x, y) = \hat{H}(x, y)$ 。对于另一半 R_L 而言，翘曲函数连续推断 \hat{H} 为相似变换 S 。由于该扭曲表现为一半进行投影变换而另一半保持形状，因此我们称之为 *保形半投影扭曲*。

3.1. 投影变换分析

我们的扭曲来自给定的投影 \hat{H} 变换分析。投影变换(单应性) $\hat{H}: (x, y) \mapsto (x', y')$ 定义为一种具有齐次坐标的线性变换, 即:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \sim \begin{bmatrix} \hat{h}_1 & \hat{h}_2 & \hat{h}_3 \\ \hat{h}_4 & \hat{h}_5 & \hat{h}_6 \\ \hat{h}_7 & \hat{h}_8 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \quad (1)$$

\sim 处表示平等到比例因子。它有八个参数。 (x, y) 和 (x', y') 之间的映射可以重新写为

$$x' = \frac{\hat{h}_1 x + \hat{h}_2 y + \hat{h}_3}{\hat{h}_7 x + \hat{h}_8 y + 1}, \quad y' = \frac{\hat{h}_4 x + \hat{h}_5 y + \hat{h}_6}{\hat{h}_7 x + \hat{h}_8 y + 1}. \quad (2)$$

我们对 \hat{H} 的分析依赖于坐标的改变。这种改变简化了公式并且揭示了重要的性质。Chum 等人[4]在单应性上计算几何误差采用了相同的技术。特别地, 我们旋转最初的坐标系 (x, y) 来形成一个新的坐标系 (u, v) 。新的坐标系 (u, v) 和原始的坐标系 (x, y) 有如下关系:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \quad (3)$$

$$\theta = \text{atan2}(-\hat{h}_8, -\hat{h}_7). \quad (4)$$

坐标系改变之后, 我们的得到一个新的从 (u, v) 映射到 (x', y') 的单应性 \hat{H} 。相应的公式变为:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \sim \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ -c & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \quad (5)$$

$$\text{where } \begin{bmatrix} h_1 & h_2 \\ h_4 & h_5 \end{bmatrix} = \begin{bmatrix} \hat{h}_1 & \hat{h}_2 \\ \hat{h}_4 & \hat{h}_5 \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}, \quad (6)$$

$$(h_3, h_6) = (\hat{h}_3, \hat{h}_6), \quad (7)$$

$$c = \sqrt{\hat{h}_7^2 + \hat{h}_8^2}. \quad (8)$$

新的映射也可以写为:

$$x' = H_x(u, v) = \frac{h_1 u + h_2 v + h_3}{1 - cu}, \quad (9)$$

$$y' = H_y(u, v) = \frac{h_4 u + h_5 v + h_6}{1 - cu}. \quad (10)$$

我们也将映射表示为 u 和 v 的函数, 即:

$$[x', y']^T = H(u, v) = [H_x(u, v), H_y(u, v)]^T \quad (11)$$



(a)原始图像 (b)投影扭曲 (c)我们的扭曲

图 2. 投影扭曲与我们的保形半投影扭曲比较, 注意到我们的扭曲比投影扭曲呈现出较少的形状和大小失真。

坐标的变化的好处是, 新的投影变换 H 具有其 $h_8 = 0$ (公式 5)。同样地, 注意到现在只有一个坐标 u 在等式 9 和 10 的分母中。坐标的变化导致了以下特性:

(i) **变更尺度**。我们首先分析了在投影变换下图像的尺度变化。这种分析依赖于矩阵分解。具体的说, H 可以如下[3]被分解为一个仿射变换 H_A 和一个纯投影变换 H_P :

$$\underbrace{\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ -c & 0 & 1 \end{bmatrix}}_H = \underbrace{\begin{bmatrix} h_1 + ch_3 & h_2 & h_3 \\ h_4 + ch_6 & h_5 & h_6 \\ 0 & 0 & 1 \end{bmatrix}}_{H_A} \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -c & 0 & 1 \end{bmatrix}}_{H_P}. \quad (11)$$

局部变化 (u, v) 可以通过 H 在 (u, v) 的雅可比行列式的测量, $\det J(u, v)$, 这与 H_A 和 H_P 的雅可比矩阵的决定因素有关:

$$\begin{aligned} \det J(u, v) &= \det J_A(u, v) \cdot \det J_P(u, v) \\ &= s_A \cdot \frac{1}{(1 - cu)^3}, \end{aligned} \quad (12)$$

$s_A = \det J_A(u, v)$ 是 u 和 v 的无关常数。从等式 12, 由 H 引起的局部区域变化的缩放系数只依赖于 u 。具体地说, 随着坐标 u 变大 H 局部面积变大, 从而导致更大的区域失真。图 2 (b) 给出了一个投影变换, 其中该区域失真沿 u 正轴更大的一个例子。

(ii) **H 的线性**。仿射变换保留直线上的长度比率而投影变换一般不这么做。然而, 坐标的变化表明, 在某些条件下, 在投影变换下长度比率也可以被保存。从等式 9 和 10 中, 如果 $u = u_0$ 是一个不变常数, 那么 $H_x(u_0, v)$ 和 $H_y(u_0, v)$ 为 v 的线性函数,

¹ 没有歧义, 我们同时在变换矩阵和其相应的映射函数上使用 H 。

$$x' = H_x(u_0, v) = \frac{h_2}{1 - cu_0}v + \frac{h_1u_0 + h_3}{1 - cu_0}, \quad (13)$$

$$y' = H_y(u_0, v) = \frac{h_5}{1 - cu_0}v + \frac{h_4u_0 + h_6}{1 - cu_0}. \quad (14)$$

换句话说，对于平行于 v 轴的线， H 保留位于线之间两点的距离比率。

(iii) **直纹曲面**。直纹曲面是由移动线扫出的面。它的参数化形式是 $\mathbf{s}(u, v) = \mathbf{p}(u) + v \cdot \mathbf{r}(u)$ ，其中 $\mathbf{p}(u)$ 被称为基线而 $\mathbf{r}(u)$ 被称为主线[5]。考虑形如 $k(u, v) = f(u)v + g(u)$ 的函数 k ，其图(由 $(u, v, k(u, v))$ 在 \mathbb{R}^3 上形成的表面)可以写成如下：

$$\begin{bmatrix} u \\ v \\ k(u, v) \end{bmatrix} = \underbrace{\begin{bmatrix} u \\ 0 \\ g(u) \end{bmatrix}}_{\mathbf{p}(u)} + v \underbrace{\begin{bmatrix} 0 \\ 1 \\ f(u) \end{bmatrix}}_{\mathbf{r}(u)}, \quad (15)$$

表明了这是一个直纹曲面。由于 $H_x(u, v)$ 和 $H_y(u, v)$ 可以写成 $f(u)v + g(u)$ 的形式，其图形是直纹曲面。公式 13 和 14 也表明了这一事实。公式 13 表明 H_x 的曲线图与平面 $u = u_0$ 的交点是一条线，这表明 H_x 的曲线图可以通过移动和旋转沿 u 轴的线形成。因此， H_x 形成了直纹曲面。根据公式 14 同样的论点也适用于 H_y 。

3.2. 半投影扭曲

我们的扭曲的建造利用了三个以上坐标变换的投影变换 H 。首先，我们用直线线 $u = u_1$ 将 R_2 分为两半， $R_H = \{(u, v) | u \leq u_1\}$ 和 $R_L = \{(u, v) | u > u_1\}$ 。对于 $(u, v) \in R_H$ ，我们运用原有的变换 H 。对于 $(u, v) \in R_L$ ，因为 H 导致 R_L 比 R_H 更大面积的失真，我们建议用相似变换 S 取代 H 。然而该扭曲必须保持连续，否则，会在扭曲图像沿 R_H 和 R_L 的边界处出现明显的接缝。为了构建一个连续的扭曲，我们需要分割线上对于所有的 (u, v) 有 $S(u, v) = H(u, v)$ 。注意，这样的要求不能满足任意分割线由于一般来说投影变换不是线性的，而相似变换是线性

的。幸运的是，因为 $H(u_1, v)$ 是 v 的线性函数，通过取 v 轴的平行线 $u = u_1$ ，由于特性(ii)，连续性的要求就可以实现。要求对于所有的 v 有 $S(u_1, v) = H(u_1, v)$ ， S 被唯一地确定为：

$$S(u, v) = \frac{1}{1 - cu_1} \left(\begin{bmatrix} h_5 & h_2 \\ -h_2 & h_5 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} (h_1 - h_5)u_1 + h_3 \\ (h_4 + h_2)u_1 + h_6 \end{bmatrix} \right). \quad (16)$$

注意 u_1 是扭曲的参数。我们将在第 4 节讨论如何确定它。

图 3 给出了一个例子。图 3 (a) 示出从给定的投影扭曲 \hat{H} 计算输入图像和 uv 坐标系。图 3 (c) 示出使用上述方法所构建的连续半投影扭曲。构造出了连续的(C^0)，半投射和半相似性的翘曲函数。与单纯的投影变换(图 3 的 (b))相比，我们的扭曲有更少的形状以及区域失真。然而，它在 $u = u_1$ 处引入了突然的线弯曲。更具体地讲，虽然 R_H 或 R_L 内的线都是直线，但是穿过 $u = u_1$ 分割线的线是弯曲的。为了减轻伪像，我们引入一个 H 是平滑外推的缓冲区，而不是在 R_L 使用单个的相似变换，以便使翘曲函数在 R_L 处是连续可微(C^1)并且从 H 渐变到相似变换 S 。

3.3. C^1 外推

构建一个 C^1 扭曲，我们建议将上述方法推广进一步将 R_L 划分为两个区域 $R_T = \{(u, v) | u_1 < u < u_2\}$ 和 $R_S = \{(u, v) | u_2 \leq u\}$ 如图 2 (c) 所示。 u_1 和 u_2 是翘曲函数的参数，将会在第 4 节说明。翘曲函数定义如下：

$$w(u, v) = \begin{cases} H(u, v) & \text{if } (u, v) \in R_H \\ T(u, v) & \text{if } (u, v) \in R_T \\ S(u, v) & \text{if } (u, v) \in R_S \end{cases}. \quad (17)$$

$T(u, v)$ 是中间区域 R_T 的一个逐渐改变其从 $H(u, v)$ 至 $S(u, v)$ 行为的函数。 $S(u, v)$ 为 R_S 定义的相似变换。我们通过下式来决定相似变换的参数：

$$S(u, v) = \begin{bmatrix} S_x(u, v) \\ S_y(u, v) \end{bmatrix} = \begin{bmatrix} \alpha & -\beta \\ \beta & \alpha \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}. \quad (18)$$

给定投影变换 H ，我们首先确定 u_1 和 u_2 然

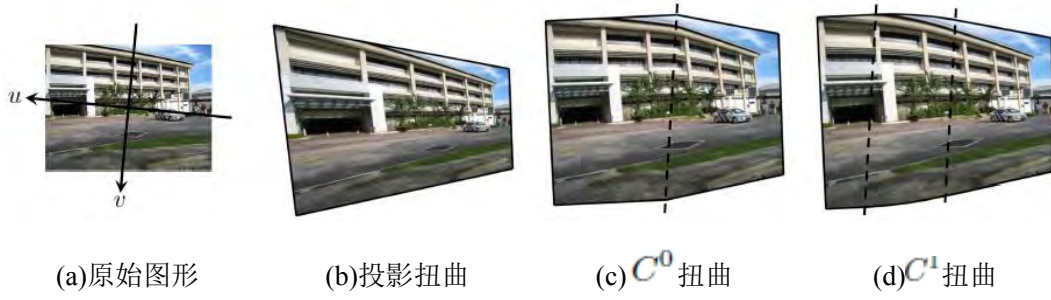


图 3 (a)原始图像 I 和从 \hat{H} 得到的 uv 坐标系。(b)用 \hat{H} 对 I 进行扭曲得到的图像。(c)用 C^0 对 I 进行扭曲得到的图像。(d)用 C^1 对 I 进行扭曲得到的图像。

后用公式 17 构造一个扭曲 w 。

根据 3.2 节特性 (iii), 我们知道图 H_x 和 H_y 都形成直纹曲面。根据公式 18, 三维平面的图 S_x 和 S_y 也是直纹曲面。这促使我们要求图 $T_x(u, v)$ 和 $T_y(u, v)$ 也是直纹曲面。因此, 我们假设 $T(u, v)$ 为以下形式:

$$T(u, v) = \begin{bmatrix} T_x(u, v) \\ T_y(u, v) \end{bmatrix} = \begin{bmatrix} f_x(u) \\ f_y(u) \end{bmatrix} v + \begin{bmatrix} g_x(u) \\ g_y(u) \end{bmatrix}. \quad (19)$$

为构建扭曲 w , 我们需要为相似变换 S 确定参数 α, β, t_x, t_y 并且为 T 确定四个函数 f_x, f_y, g_x, g_y 。我们假设这四个函数是 u 的多项式函数。将公式 9, 10, 18 和 19 代入到公式 17 中, 我们可以将翘曲函数 w 表达为以下形式:

$$w(u, v) = \begin{bmatrix} w_x(u, v) \\ w_y(u, v) \end{bmatrix} = \begin{bmatrix} F_x(u)v + G_x(u) \\ F_y(u)v + G_y(u) \end{bmatrix}, \quad (20)$$

其中

$$F_x(u) = \begin{cases} \frac{h_2}{1-cu} & \text{if } u \leq u_1 \\ f_x(u) & \text{if } u_1 < u < u_2 \\ -\beta & \text{if } u_2 \leq u \end{cases}, \quad (21)$$

$$G_x(u) = \begin{cases} \frac{h_1 u + h_3}{1-cu} & \text{if } u \leq u_1 \\ g_x(u) & \text{if } u_1 < u < u_2 \\ \alpha u + t_x & \text{if } u_2 \leq u \end{cases}. \quad (22)$$

F_y 和 G_y 可以被相似地表达。

剩下的问题就是如何寻找合适的 $f_x, g_x, f_y, g_y, \alpha, \beta, t_x$ 和 t_y 使得 w 是 C^1 连续的。这就需要 F_x, G_x, F_y 和 G_y 都是 C^1 连续的。取 F_x 为例, 根据公式 21, 若以下四个条件均成立 F_x 就是 C^1 ,

$$f_x(u_1) = \frac{h_2}{1-cu_1} \quad (F_x(u_1) \text{ is continuous}) \quad (23)$$

$$f'_x(u_1) = \frac{ch_2}{(1-cu_1)^2} \quad (F'_x(u_1) \text{ is continuous}) \quad (24)$$

$$f_x(u_2) = -\beta \quad (F_x(u_2) \text{ is continuous}) \quad (25)$$

$$f'_x(u_2) = 0 \quad (F'_x(u_2) \text{ is continuous}) \quad (26)$$

因为我们有四个线性约束, 因此我们可以得到至少四个参数。由于 β 已经有了一个参数, 所以 f_x 可以有三个参数。因此我们假设 f_x 是 u 的一个二次函数。通过求解所得的 4×4 线性系统, 我们得到 $f_x(u)$ 和 β 。同样, 可以通过对 F_y 的连续性运用 C^1 得到 $f_y(u)$ 和 α 。遵循同样的策略, G_x 上的限制给出了 $g_x(u)$ 和 t_x 。最后, G_y 给出了 $g_y(u)$ 和 t_y 。

总之, 所提出的扭曲通过自动确定函数 $T(u, v)$ 和 $S(u, v)$ 从 $H(u, v)$ 推出 R_T 和 R_S 。最后, 给定联系两个输入图像的投影变换 \hat{H} , 我们构建了一个 C^1 连续的, 一半投影而在另一半连续地转变为相似变换的翘曲函数 w 。图 3 (d) 体现了所构建翘曲。下一节将讨论如何将该扭曲运用于图像拼接, 以及确定参数 u_1 和 u_2 。

4. 图像拼接

本节介绍如何通过扭曲进行图像拼接。我们首先描述它如何用于两两拼接, 以及如何自动确定参数 u_1 和 u_2 。然后, 我们提出了一个的方案建议将所提出的扭曲与尽可能投影 (APAP) 扭曲[13]相结合从而拥有两者的优势。

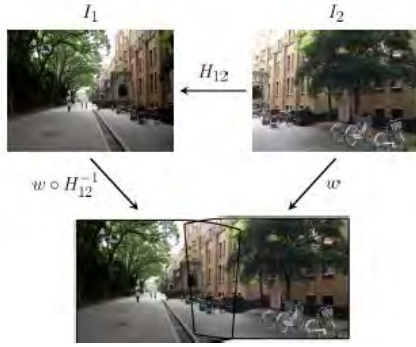


图 4.使用所提出的扭曲后两个图像之间的几何关系。

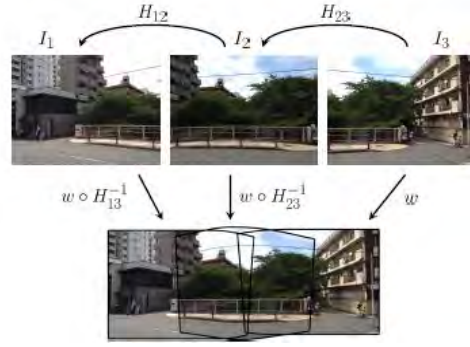


图 5.使用该扭曲后三幅图像之间的几何关系。

4.1. 保形图像拼接

给定两个图像 I_1 和 I_2 , 假设它们是由一个从 I_2 中 (x_2, y_2) 映射到 I_1 中 (x_1, y_1) 的单应矩阵 H_{12} 联系起来的。如图 4 所示, 如果 I_2 被 w 翘曲, I_1 应当被 $w \circ H_{12}^{-1}$ 翘曲, 以保持 H_{12} 所描述的几何关系。当 $w = H_{12}$ 时, 用于 I_2 的 H_{12} 以及 I_1 的固定会减少。当 $w = I$ 时, I_2 被固定并且 I_1 被 H_{12}^{-1} 翘曲。如第 3 节描述那样我们从 H_{12}^{-1} 得到了所提出的翘曲并在图 4 中把它用作 w 来拼接两个图像。

接下来, 我们描述了如何自动地确定参数 u_1 和 u_2 。由于该翘曲的一个目标是保持每个图像的角度, 我们希望每一个图像尽可能地进行相似变换。也就是说, 我们希望这样的构造的翘曲 w 趋近于相似变换。为此, 我们将每个图像 I_i 与可以在 Frobenius 范数中衡量其翘曲函数 w_i 与最近的相似变换的偏差的代价 E_i 的结合, 即:

$$E_i(u_1, u_2) = \min_{a_i, b_i} \iint_{(x,y) \in \Omega_i} \left\| J_i(x, y, u_1, u_2) - \begin{bmatrix} a_i & -b_i \\ b_i & a_i \end{bmatrix} \right\|_F^2 dx dy \quad (27)$$

其中 Ω_i 是 I_i 的矩形域; $J_i(x, y, u_1, u_2)$ 是 w_i 在 (x, y) 上计算的雅可比矩阵。由于 u_1 和 u_2 是 w_i 的参数, 所以雅可比矩阵取决于 (u_1, u_2) 。在拼接两幅图像的情况下, 总能量 $E(u_1, u_2) = E_1(u_1, u_2) + E_2(u_1, u_2)$, 其中 $w_1 = w \circ H_{12}^{-1}$ 并且 $w_2 = w$ (w 是由 H_{12} 构造的扭曲)。需要注意的是, 虽然减少个别 E_i 可能导致相似变换, 他们的联合优

化组合在校准和相似的要求之间找到了一种良好的折中方案。能量是 u_1 和 u_2 的一个非线性函数。我们通过参数空间 (u_1, u_2) 定期取样, 在采样位置上评估能量并且选择具有最低能量的取样点优化它。

对于拼接图像序列 $\{I_1, \dots, I_n\}$, 我们进行了一次性同时对所有输入图像拼接的智能分组拼接。我们的方法找到每对相邻图像的单应矩阵 $H_{i,i+1}$ 以及第一幅和最后一幅图像之间的单应矩阵 H_{1n} 。 H_{1n} 用于导出扭曲 w 。 I_n 在由 w 扭曲并且其他每幅图像 I_i 由 $w \circ H_{in}^{-1}$ 扭曲。图 5 示出了一个同时拼接三个图像序列的例子。

4.2. 与 APAP 扭曲相结合

所提出的扭曲更多的关注维护角度而不是对准精度。然而, 它可以与具有更好的对准精度的方法进行组合从而具有两者的优势。APAP 扭曲[13]是在对准精度方面最先进的图像校准方法之一。一种可能与 APAP 结合的方式是在移动 DLT 框架中所提出的半投影扭曲取代投影扭曲。然而, 这需要确定空间变化 (u_1, u_2) , 这是不常用且不直观的。在下文中, 我们描述了将 APAP 扭曲与所提出的扭曲结合起来的简单方案。

我们解释我们的扭曲为两阶段的过程: 如图 6 (a) 中, 在快速扭曲 $w \circ H^{-1}$ 之后进行一个投影变换 H 。这种快速扭曲可以被视为减少投影失真的一个细化。具体来说, 我们的快速扭曲只处理 R_L 中有大面积变形的地方, 而不对 R_H 进行处理。尽管 APAP

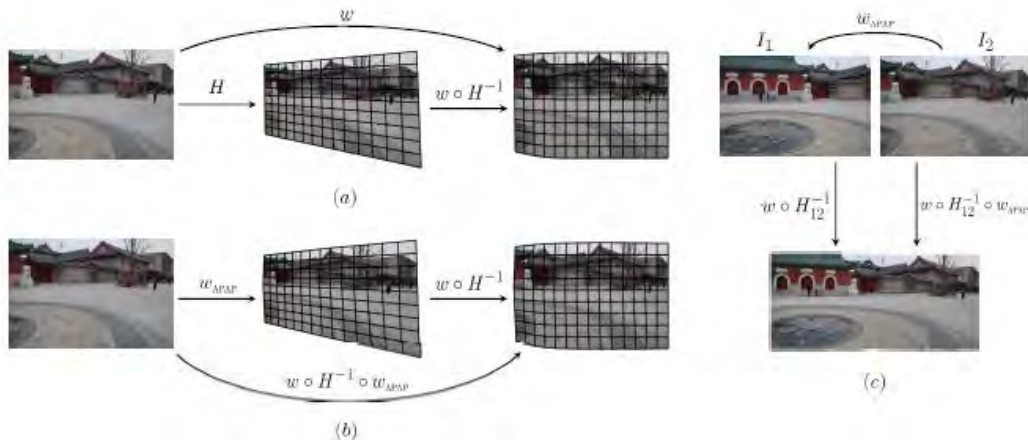


图 6.(a)我们的扭曲可以被解释为一个快速扭曲后经过细化的投影扭曲。(b)在 APAP 扭曲之后应用我们的快速扭曲的整合。(c)使用 APAP+我们的扭曲后的两幅图像之间的几何关系。



图 7.对比投影扭曲，自动拼接和我们的扭曲

扭曲是局部变化的扭曲，它也遵循全局地投影变换。因此，我们可以简单地把它作为一个投影扭曲然后在其上应用我们的快速扭曲以减少失真，如图 6 (b) 所示。以这种方式，将图像通过 APAP 局部对齐，而全局形状通过我们的快速扭曲调整。概括地说，将 APAP 与快速扭曲相结合后的翘曲函数可定义为

$$w \circ H^{-1} \circ w_{APAP}. \quad (28)$$

图 6 (c) 表示采用 APAP 和我们的扭曲的组合时更新后的几何关系。

5. 实验

我们将该扭曲与投扭曲和自动拼接[1]



图 8.利用自动拼接，投影扭曲和我们的扭曲拼接 10 幅图像的例子比较。

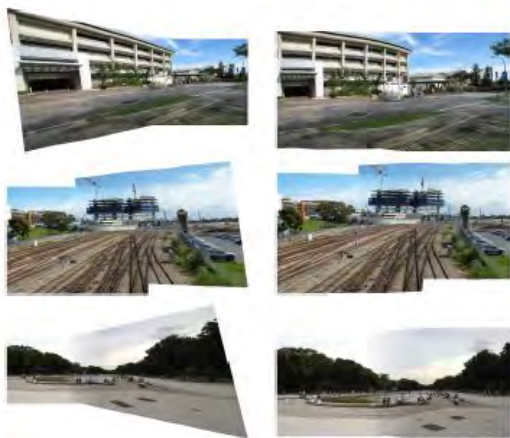


图 9. APAP 扭曲与 APAP 和我们的扭曲的组的比较。

比较。自动拼接假定输入图像共享相同的照相机中心，并使用 3D 旋转模型。对于投影扭曲和我们的扭曲，我们采用 VLFeat 库[12]来提取 SIFT 特征并且通过与 RANSAC 匹配来获得重叠区域的投影变换。通过固定一幅图像作为参考，并在其他图像上使用投影变换来实现利用投影扭曲的拼接。为得到具有更少失真的最好的投影扭曲我们手动选择参考图像。对于我们的扭曲，在获得投影变换后，我们确定参数 u_1 和 u_2 ，构造扭曲，然后运用翘曲函数来对齐图像。经对准的图像使用线性混合来合成。

我们在 MATLAB 上执行了所提出的方法并且在配有 3.4GHz 的 CPU 和 4GB 内存的 PC 上做了实验。对于拼接 800x600 分辨率的两幅图像，我们用了大约 5 秒来查找参数 u_1 和 u_2 ，以及小于 2 秒的时间来构造扭曲和图像变组合。

图 7 比较了投影扭曲，自动拼接和我们

的扭曲。投影扭曲保留所有直线条，但表现出在大小，形状和方位的投影失真。自动拼接可以减少面积失真。但是，它弯曲了一些重要的直线。我们的扭曲减轻投影失真，同时保持各图像的视角。因为无论投影还是相似变换都不会引入线失真，所以我们的扭曲只在 R_T 上会有线失真。图 8 在拼接十幅图像序列上比较了这三种方法。自动拼接在本例引起了严重重的形状失真。投影扭曲有面积和形状畸变尤其在两端（孩子的形状被扭曲）。我们的扭曲有较少的失真，并提供了所述场景的多维视角。

图 9 在两幅图像的拼接上比较了 APAP 扭曲和 APAP 与我们的扭曲的组合。我们的快速扭曲是通过如图 6 (c) 中所述的扭曲在两个图像上执行。因此，我们的扭曲可以像 APAP 那样保持对准精度同时调整结果的整体形状来减少失真。需要注意的是，这些例子不能通过单个投影变换很好的对准。通过与 APAP 的结合，所提出的扭曲变得更稳健。

6.结论

本文为图像拼接提出了一种新颖的扭曲。我们的扭曲全面地校准图像，同时保留其原有的视角。当与 APAP 扭曲结合时所得的结果可以提供精确的校准，更小的失真和多维视角。我们的参数选择的过程目前不将图像内容（例如，线功能）考虑在内。因此，当场景是满线结构的，所选择的参数在减少线失真上是无效的。

今后，我们还要探索将我们的扭曲运用到不同的场合如视图变形的可能性。

致谢 这项工作的部分是 Che-Han

Chang 作为东京大学的访问学生, 在交流协会日本(IAJ)的支持下完成的。这项工作被部分被授予 NSC101-2628-E-002-031-MY3 和 NSC102-2622-E-002-013-CC2 所支持。

参考文献

- [1] M. Brown, D. G. Lowe 利用不变特征自动全景图像拼接, *国际计算机视觉期刊*, 4(1):59–73, 2007
- [2] R. Carroll, M. Agrawal, A. Agarwala 优化广角图像的内容保留的投影, *ACM 图形学汇刊*, 28(3):43, 2009
- [3] O. Chum, J. Matas 平面变化尺度仿射纠正, *2010 年 ACCV 学术会议*, 347–360 页, 2011
- [4] O. Chum, T. Pajdla, P. Sturm 单应矩阵的几何误差, *计算机视觉和图像理解*, 97(1):86–102, 2005
- [5] M. P. Do Carmo 曲线和曲面的微分几何, *培生出版社*, 1976.
- [6] J. Gao, S. J. Kim, M. S. Brown 采用双应性变形构造图像全景, *2011 年 IEEE CVPR 会议记录*, pages 49–56, 2011
- [7] R. I. Hartley, A. Zisserman *计算机视觉中的多视图几何*, 剑桥大学出版社, 第二版, 2004.
- [8] J. Kopf, D. Lischinski, O. Deussen, D. Cohen-Or, 和 M. Cohen 减少全景畸变的局部适应预测, *计算机图形学论坛*, 28(4):1083–1089, 2009
- [9] W.-Y. Lin, S. Liu, Y. Matsushita, T.-T. Ng, L.-F. Cheong 平滑的变仿射拼, *2011 年 IEEE CVPR 会议记录*, 345–352 页, 2011
- [10] R. Szeliski 图像校准与拼接:教程, *计算机图形和视觉*, 2(1):1–104, 2006.
- [11] R. Szeliski, H.-Y. Shum 创建全景图像拼接技术和环境映射, *1997 年 ACM SIGGRAPH 会议记录*, 251–258 页, 1997
- [12] A. Vedaldi, B. Fulkerson. VLFeat 开放便携式计算机视觉算法库, *2010 年 ACM 多媒体会议记录*, 1469–1472 页 2010
- [13] J. Zaragoza, T.-J. Chin, M. S. Brown, D. Suter 尽可能地投影图像拼接与移动的 DLT, *2013 年 IEEE CVPR 会议记录* 2339–2346 页 2013
- [14] L. Zelnik-Manor, G. Peters, P. Perona, 解决全景难题, *2005 年 ICCV 会议记录* 第二卷 1292–1299 页 2005