# 评估----2Dprojective transformation

本节中，我们开始考虑评估问题。包括主要内容：一些transformation的计算，或者mathematical quality。

Estimation problem分类：

（1）**2D homography：**在P2中给定一系列points Xi，在类似的P2中给定一系列对应的points X’i，计算projective transformation让Xi与X’i对应起来。实际生活中，一般Xi与X’i 在两张不同的图片上，每张图片都对应一个projective plane P2。

（2）**3D到2D的camera projection：**在三维空间中给定一系列points Xi，在一张图片中给出一系列对应的points xi，找到3D到2D的projective mapping 使得Xi映射到xi。这样的3D到2D的mapping是由projective camera计算出来。

1. **fundamental matrix computation：**给定一张图片中的一系列points xi，给定另一张图片中对应的一系列points x’i，联合计算出fundamental matrix F，F是3\*3的singular matrix，满足x’iT F xi=0。
2. **Trifocal tensor computation：**给定三张图片中一系列points xi及其对应点x’i,x’’i，计算trifocal tensor。

上面的estimation problem具有共性，解决了一个就可以解决其他的。

Estimating 2D projective transformation问题是很重要的，我们考虑两张图片中的一系列points xi和其对应point x’i，我们的目标是计算出一个3\*3的matrix H，使得Hxi=x’i

**需要measurement的数量：**第一个问题是计算projective transformation H需要多少个correspond points xi---x’i，主要考虑自由度的数量和constrain数量的下届。一方面，matrix H包含9个entries，这样2Dprojective transformation的自由度为8。另一方面，每一个points-to-points correspondence代表两个constrain。2D中的点有两个自由度（x和y），即使被写为homogeneous 3 vector形式，其也有2个自由度。为了完全constrain H，有必要选出4对point correspondence。

**Approximate solution：**如果给定了4对correspondence，则有可能求出H的exact solution。这是minimal solution。 H的minimal solution很重要，因为它定义了robust estimation algorithm所需求的子集大小。然而，因为noise的存在，point取的也不准确，则可能计算不出fully constrain的projective transformation，其中一个将要面对的任务是从给定的data中计算出最好的transformation。解决这个任务的一般方法是找到使cost function最小的transformation H。主要有两类cost function：基于minimizing algebraic error，和基于minimizing geometric（statistical image distance）。

**The gold standard algorithm：**通常会有一个cost function是最优的，因为minimizing H的过程给出了transformation的最佳可能estimation，这个能minimizing cost function的方法叫gold standard algorithm。其他algorithm是通过与gold standard比较来进行评估。

## DLT算法 direct linear transformation algorithm

开始计算H。Transformation等式为x’i=Hxi，这个等式中的x是homogeneous vector，这导致3-vector x’i和Hxi不对等，他们有相同的方向，但是非零scale factor的尺度大小可能会不一样。Transformation等式可以写为向量叉乘的形式，这个公式可以计算出H的简单线性解。H的第j行用表示，HXi可以写成如下形式



，则可展开为：



因为，则上面式子进一步写为

 （4.1）

上式可以写成，Ai是3\*9的矩阵，h是9\*1的向量，h展开就是H

 （4.2）

上面的推导有三个点需要理解：

1. 是一个线性等式，其中h未知，而矩阵Ai的元素是已知point的坐标的quadratic。
2. 尽管公式（4.1）中有三个等式，但只有两个是线性独立的（因为第三行是第一行和第二行的和得到的），所以在求解H时，每个point correspondence都会得到两个等式，第三个等式通常会被忽略。所以（4.1）可以写成如下形式：

 （4.3）

此时Ai是2\*9的矩阵

1. 该等式适用于任何用homogeneous coordinate表示的point。c’i=0意味着（a’i，b’i）是图片上的坐标。

H的解

每个point correspondence给出了关于H的两个独立等式，给定4个point correspondence，我们获取了一系列类，A是equation coefficient的矩阵，A的每一行Ai由correspondence组成，h是未知实体H转换成的向量。我们寻找h的非零解（我们不感兴趣h=0）。如果使用的是公式（4.1），那么A是12\*9的矩阵，如果使用的公式（4.3），那么A是8\*9的矩阵，无论是哪种情况，A的秩都是8，所以一个一维的null-space提供了h的解。

这样的h只能被一个non-zero scale factor确定。然而，H通常取决于scale，scale可能被h的norm选择，比如。

### 超定解

如果给定了超过4个point correspondence，则一系列的得到的解是超定。如果point的位置是确定的，则矩阵A的秩为8，一个一维null-space，此时h的解是精确的。但实际情况并没有这么理想，如果图片中的坐标并不精确，那么的over-determined system并不能得到一个精确解。因此人们尝试用近似解来提到精确解，也就是求使cost function最小时向量h的解。为了避免h解出来为0，需要额外的constrain，一般来说使用的是对norm设置条件，比如。Norm的值很重要，因为H只由scale决定。假定没有精确解，那么用最小化norm来替代，同时其服从constrain。这等同于找到的最小值。在5.3节中将介绍的最小eigenvalue对应的eigenvector就是h的解。同样的，这个解也是A的最小singular value对应的singular vector。这个计算方法也就是基础DLT算法。

### 非齐次解

另一个h的解法是将一系列（4.3）等式转为非齐次线性等式，这这方法需要对向量h加入限制条件加入限制条件是合理的，因为h的解只和scale相关，这个scale可以被比如选择出来。例如，假设h的最后一个元素（对应的是H33）被选为unity，则（4.3）等式可以被写为



是长度为8 的向量，对应的是h的前8个元素。把四个correspondence联立起来生成一个matrix equation ，M有8列，b是8维向量。这样的等式是可以用解线性等式的standard technique解出h，其中M有8行，或者通过least-square techniques得到等式的超定解。然而，事实上是真正的解，这样就不存在乘数比例的k使得，这意味着我们无法得到true solution。为了解决这个问题，可以将该方法引导到求解一个unstable result，即令hj接近0.这种方法通常不被推荐。

使用这种方法，当一个point被H映射到infinity，则会有=0。因为表示坐标原点，也表示infinity line l，这个条件可以被写为，这样H33=0。在scene plane的perspective image中infinity line被投射为vanishing line，例如，ground plane中的地平线就是vanishing line。地平线穿过image center（可以令坐标原点和image centre重合）并不少见。在这种情况下，world plane的图片将origin映射到infinity line，这样会得到true solution H33=h9=0。最终，h9=1进行归一化将无法得到想要的解。

### degenerate configuration

计算最小解所使用4个point correspondence要是homograph，并假设3个point x1，x2，x3共线。如果correspondence point x’1，x’2，x’3也是共线，那么可以推断homograph没有得到充分的约束，并存在一个xi到x’i的homograph映射家族。另一方面，如果correspondence point x’1，x’2，x’3不共线，则不存在transformation H将xi映射到xi’，因为projective transformation必须保持共线。然而公式（4.3）的8个homogeneous等式必须有一个非零解H，如何解决这个明显的矛盾呢？

公式（4.3）表达的条件，所以矩阵H可以由满足条件的8个等式解出来。假设，x1...x3共线，该条线记为l，则有。现在定义，其中l是3\*3的矩阵2。另一方面，，因此公式对于所有的i都是满足的。注意通过公式可以将向量h\*和H\*对应起来，且向量h\*满足公式（4.3）。问题是H\*是rank为1的矩阵，这样的H\*并不能表示为一个projective transformation。

我们已经知道如果x1，x2，x3共线，则得到公式（4.1）的解。这里会产生两种情况：要么H\*是唯一解，要么存在一个further solution H。第一种情况，因为H\*是singular matrix，所以不存在一个transformation将xi映射到xi’；这种情况一般发生在x1...x3共线，而x1’....x3’不共线的情况。第二种情况，当存在一个further solution H时，任何形如的矩阵都是一个解。这样就存在一个transformation的二维家族，推导出这个transformation的8个等式来自于公式（4.3），且这8个等式不是线性独立的。

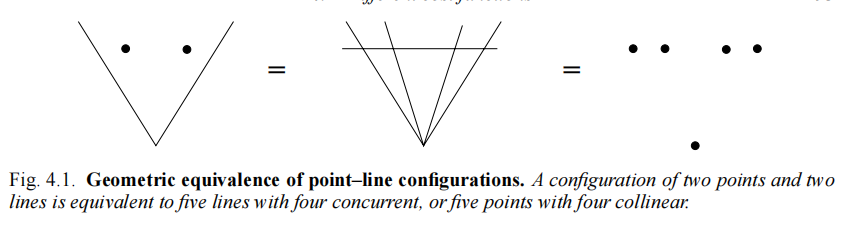
有一种情况不能解出transformation的唯一解，这种情况叫degenerate。注意定义一个degeneracy同时涉及transformation的type和configuration。然而，degeneracy问题并不是limit来得到最小解。如果提供额外的共线对应关系，则degeneracy将不能被解出来。

### solution from lines and other entities

本节剩下部分主要集中于计算point correspondence的homographies。然而，通过line correspondence也可以给出一种确定homographies。以line transformation 为起点，可以推导出公式，要想求出该公式的最小解，要求4条线在general position。类似的，homography也可能通过conic correspondence求出来。

现在引入新的问题——计算homography需要多少correspondence？通常的规则是constrain的数量必须大于等于transformation的自由度。例如，在2D中，每个correspondence point或者line correspondence都会为H生成两个constrain；在3D中，每个correspondence point或者line correspondence都会为H生成3个constrain。所以在2D中，要满足计算出H需要4个point correspondence或者4个line correspondence；在3D中，homography有15个自由度，因为需要5个point correspondence或者5个line correspondence。而对于一个planar affine transformation，仅仅需要3个point correspondence或者3个point correspondence。一个conic可以为2D的homograph提供5个constrain。

在使用混合类型的correspondence计算H时需要尤其小心。例如，2D中，2个point correspondence再加2个line correspondence不能唯一确定homography，但是3个point correspondence再加1个line correspondence或者1个point correspondence再加3个line correspondence可以唯一确定唯一homograph。3个point correspondence再加1个line correspondence等价于4个point correspondence，因为3个line correspondence定义出一个triangle，且triangle的三个顶点对应3个唯一的point。我们已经看到general position中的4个point correspondence可以唯一确定homograph，这意味着3个point correspondence再加1个line correspondence也可以唯一确定homograph。类似的，1个point correspondence再加3个line correspondence对应4个line correspondence，而4个line correspondence可以唯一确定homograph。然而，图4.1中展示了，2个point correspondence再加2个line correspondence等价于交于4个点的5条line，或者对应5个点其中4个点共线。在前几节已经介绍过，这种配置是degenerate，并且homograph的one-parameter family会将two-point和two-line的configuration映射到对应的configuration。



## 不同的损失函数

本节将描述不同的损失函数，以求最小化损失函数来确定H的over-determined solution。损失函数的最小化方法将在下面给出

### algebraic distance

DLT算法最小化norm，向量叫做residual vector，这是error vector的norm。这个向量的组成来自于individual correspondence，individual correspondence对应矩阵A的每一行。每个correspondence xi <--> xi’贡献一个partial error向量。这个向量是与correspondence xi <--->xi’和homograph H相关联的algebraic error vector。这个向量的norm是一个scalar，这个scalar我们称之为algebraic distance：



将上式一般化，对于任何两个向量x1和x2，我们可以得到如下公式：

其中

这个距离和几何距离之间的关系将在4.2.4节中说明。

给定一系列correspondence，是complete set的algebraic error vector



Algebraic distance 的理论源自于conic-fitting。但是algebraic distance的缺点是大量的最小化在几何上和统计上是无意义的。最小化algebraic distance的解可能不是直觉上预测的那样。无论如何，最小化algebraic distance 仍然是normalization 方法中的一个不错的选择。Algebraic distance的特殊优势是其解释线性解，算力低。大多数基于algebraic distance的解都被用来作为求非线性的最小化几何和统计损失函数的起点。非线性最小化给出的解有一个最终的“polish”。

### geometric distance

接下来我们将讨论基于图像中几何距离的alternative error function，并最小化测量坐标和图像估计坐标之间的差距。

**注意：**向量x表示measured image coordinate；表示point的estimated value，表示point的true value。

Error in one image：当第一个point被准确measured后，我们只考虑出现在第二张图片中的error。但是这在大多数实际情况下是不可能出现的。假设一个更加合理的例子，估计calibration pattern和world plane之间的projective transformation，在估计的过程中，image力的point的measured值是很准确的。此时适当数量的transfer error需要被最小化。这是第二张image中measured point x’和point 之间的euclidean image distance，其中对应的point 被映射到第一张image中。我们使用符号来表示inhomogeneous point x和y 之间的euclidean distance。这样一系列correspondence的transfer error可以表示成如下形式：

 （4.6）

Estimated homography 是（4.6）H中的一个

对称transfer error：在大多数实际情况中image measurement error在两张图像中都会发生，因此最好是将两张图片的error都最小化。一种更好的策略是考虑forward（H）和backward（）的error function，并计算两个对应transformation之间的geometric error之和。可以写为如下形式：

 （4.7）

求和的第一项时第一张图片中的transfer error，求和的第二项是第二张图片的transfer error。

### 两张图片中的reprojection error

另一种量化两张图片error的方法涉及评估每个correspondence的correction。一个问题是通过correct每张图片中的measurement来达到获取精确image point 的目的到底存在多少必要性。 需要将这个和（4.6）的几何transfer error进行比较，为了得到准确的matching point，每张图片中对测量结果进行调整是很有必要的。

在目前的情况中，我们正在通过最小化total error function来得到homography 和精确匹配的 ：

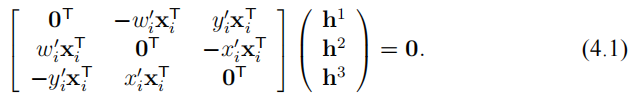
 （4.8）

最小化这个cost function需要同时确定和附带的correspondence  。这个estimation model可以测量图片中的point对应着world plane中的point。我们希望从估计出world plane中的point ，然后reprojected来精确估计出matched correspondence 。

在图4.2中reprojection error function和symmetric error function进行了比较。在后面的4.3节中我们将再次见到其与 homography和correspondence的maximum likelihood estimation相关。

### reprojection error的几何解释

Estimate两个plane之间的homography可以被看做是在4D空间中拟合一个surface。图片中每对point x，x’，在space R4中定义一个single point， 用X表示，inhomogeneous coordinate x和x’联系起来就形成了X。对于一个给定的homograph H，image correspondence满足，定义一个algebraic variety ，是两个quadric hyper surface的交点。R4中的surface是一个quadric，因为（4.1）中的每一行是一个关于



x,y,x’,y’的二次多项式。H的元素决定了polynomial每一项的coefficient，所以H确定了particular quadric。（4.1）中两个独立的等式定义了两个这样的quadric。

在D R4中给定point ，则估计homography的任务转变为找到一个经过point xi的variety 。通常来说不太可能精确的拟合出一个variety。既然这样，那就让成为对应transformation H的变种，对于每一个point xi，让成为variety 上最靠近point xi的存在。上面的描述可以写为：



这样R4中的geometric distance等同于两张图片中的测量reprojection error，找到variety 和位于上的point ，最小化distance的square sum等同于找到homography 和通过最小化reprojection error function得到estimated point和

在上的point是距离measured point X最近的点，位于X和之间的线垂直于在的切平面。从而得到公式：



其中代表point X到variety 的垂直距离。在下面的章节中将讨论conic-fitting analogue，conic-fitting analogue会说明会存在多个类似从X到的垂直距离。在R4的rigid transformation中 是不变的，其包括坐标（x,y），(x’,y’)之间的rigid transformation。

Conic analogue：接下来将analogous estimation problem可视化。问题是将conic拟合到2D point，这个问题在拟合straight line和拟合homograph中占据一个很重要的位置。

将一个conic拟合到一组平面上大于5个点的point组的方法是基于最小化geometric distance。Point可能被看做是correspondence 。在图4.3中解释说明了什么是transfer distance和reprojection distance。从图中可以得到结论小于等于transfer error。

来自conic C的Point x的algebraic distance可以通过公式定义。C的线性解可以从最小化得到。并不存在一个线性表达式来表示point(x,y)到conic C的垂直距离，因为R2中的每一个点都最多有4条线垂直于C。但是可以将function定义为返回point到conic之间的最短距离。一个conic可以通过最小化来estimate，尽管不存在线性解。给定一个conic C和一个measured point x，通过选择C上最近的point的方法可以计算出correct point 。

现在我们重新关注估计homograph。在affine transformation中variety是两个hyperplane的intersection。例如2维空间的linear subspace。公式（4.9）中在x,x’,y和x,y,y’之间产生一个线性约束。一个类似的情况是将line拟合到plane的point上。这两种情况下都通过最小化point和variety之间的垂直距离还estimate。在这两种情况下，都有一个封闭形式的解决方案。

### 4.2.6 Sampson error 桑普森错误

公式（4.8）中的geometric error非常复杂，并且最小化geometric error要求同时估计出homograph和point  。这个非线性的估计问题将在4.5节中介绍。与简单的最小化algebraic error相比，最小化geometric error是复杂的。Geometric error的geometric解释在4.2.5节给出，geometric error进一步推出cost function，该cost function位于algebraic cost function和geometric cost function之间，但是geometric error给出的结果实际上是接近geometric error。我们将这种cost function称为Sampson error。

在4.2.5的描述中，最小化geometric error得到的vector 是最近variety的point。这个point不能直接通过迭代来估计，variety是非线性的。Sampson error function的理论是估计point 的 first-order approximation，假设cost function能够在临近estimated point附近能够很好的线性逼近。接下来的讨论中心与2D homograph estimation problem相关，但基本上不适用于本书中讨论的其他估计问题。

对于一个给定的homography H，在上的任意point 是满足等式（4.3-p89），或者。为了加强对于X的依赖，我们将前面的等式用，是2维向量。这个cost function可以使用Taylor 展开式逼近：

 （4.10）

其中，在variety 上，进而推导出，后面我们将简化维，其中J是偏导数矩阵。我们如今面临的找到最小值的问题，可以简化为如下等式：

找到使最小的vector ，其中

解决这个问题的标准方法是使用Lagrange multiplers。Lagrange multiplers的一个vector被引入进来，问题也被简化为寻找的极值，其中乘数2只是为了方便而引入的。关于vector 的derivative，通过令下式为0得到：



从上式我们可以得到，的derivative给出了的original constraint，替换得到：



而

进而得到

因为则SamPOSon error 的表示——norm可以计算为：



**线性cost function：**

Algebraic error vector是典型的多线性。其中是线性的。第一个需要注意的点是Taylor展开式给出的对geometric error 的first-order approximation是精确的，这意味着Sampson error和geometric error是相同的。

此外，通过等式定义的variety 是一系列线性等式，依赖于hyperplane H。找H的问题进一步演变为hyperplane fitting problem---即在由H参数化的超平面中，找到对数据Xi的最佳拟合。

上述思想的一个例子是公式（4.8）中最小化affine transformation的geometric error

### 4.2.7 另一个geometric 解释

Homograph可以将一系列points xi 和一系列point 对应起来，在4.2.5中将寻找homograph的问题和在R4中通过一系列point拟合一个variety的问题等同起来。我们现在考虑另一种不同的解释，该解释是在RN空间中，将一系列measurement用一个是single point 表示。前提是所有的estimate problem都要符合同一个framework。在abstract term中estimate problem由两个部分组成：

1. measurement space RN由measurement vector X组成
2. Abstract term中的model被简单看做是RN中point的子集。子集中的Measurement vector X被认为是满足model的。通常满足model的subspace被认为是一个submanifold 或者是RN中的variety。

现在给定RN中的measurement vector X，则estimation problem被看做是找到最接近X的vector ，且其满足model。接下来将介绍如何使用这个framework解释2D homograph estimation problem 。

**Error in both image**：令表示一系列measure matched point。这样就有4n个测量值，即在两张图片的两个坐标系中一共有n个point。这一系列matched point在RN中可以用一个point 表示，也就是N=4n。由两幅图像中所有匹配点的坐标组成的向量将记为X。

当然并不是所有的point pair 都可以通过homograph对应起来。一系列point correspondence是可以通过projective transformation H（,其中所有的i都构成了RN的子集）对应起来，则这些point correspondence满足model。通常来说，这些point两形成RN中的submanifold S。这个submanifold的维度等同于用于parametrize submanifold的最小parameter数量。

在第一张图片中任意选择n个point。此外，还可以任意选择 homograph H。一旦选定后，第二张图片中的也就通过确定了。这样，2n+8个参数的选择导致选择的灵活性：2n表示point 的坐标，8表示transformation H的8个独立参数。这样submanifold 有2n+8个维度，和2n-8个hence codimension。

给定一系列measured point pairs，和RN中对应的一个point X，以及在S上的estimated point ，就可以很容易证明



这样，找到RN中最接近于X的位于S上的等同于最小化cost function（4.8）。Estimated correct correspondence于RN中最接近于surface point 对应，因此一旦知道了，H也就可以被计算出来。

Error in one image ONLY：单张图片中出现的error是一系列correspondence，假定是完美的，的非齐次坐标组成了measurement vector X。在这种情况下，measurement space具有维度N=2n。由完美匹配的非齐次坐标组成的vector 是随着H的变化而变化的满足model的measurement vector 的集合。Subspace是一个variety，其维度是8，因为这是homograph matrix H的自由度总数。与前面的情况一样，codimension为2n−8。这可通过下式证明



因此，在S上找到离measurement vector X最近的点等价于最小化代价函数（4.6）

## statistical cost function 和 maximum likelihood estimation

在4.2中，各种cost function都是计算的图片中的estimated point 和measured point之间的几何距离。使用这种cost function现在是合理的，然后通过统计图中measured point的误差来泛化这种cost function。

为了获得最好的H，确定model的measurement error是很有必要的。我们在这里假设，在没有测量误差的情况下，true point满足homography，例如。一个共同的假设是图像坐标测量误差服从高斯（或正态）概率分布。这种假设在一般来说肯定是不合理的，并且不考虑measured data中的异常值的存在。检测和去除异常值的方法将在后面的第4.7节中进行讨论。移除异常值后，在没有严格证明的情况下，高斯（或正态）概率分布会更加合理。因此，对于目前，我们假设图像测量误差服从zero-mean isotropoc gaussian distribution。这个分布在A2.1节中都有描述。

我们假设noise是每个图像坐标上均值为零，标准差一致为σ的高斯分布。这意味着其中服从方差为的高斯分布。如果进一步假设每个测量值上的噪声都是独立的，并且true point 是，则每个measured point x的probability density function表示为如下公式：



**Error in one image：**首先，我们考虑误差只出现在第二幅图像中的情况。Probability可以简单的从correspondence中各自的probability density function得到，因为每个point的error都是单独累加的。则noise-perturbed data的probability density function可以写为如下公式：



符号表示给定homography H后获得的概率。Correspondence set的log-likelihood的公式写为：



Homography 的maximum likelihood estimate（MLE），最大化这个对数，因此，我们注意到ML估计相当于最小化几何误差函数（4.6）。

Error in both image：遵循与上述类似的发展，如果true correspondence是，则noise-perturbed data的probability density function可以写为：



此外，我们必须找到“校正后”的图像测量值。这样Projective transformation H和correspondence的ML estimate是homography和correspondence的minimize，公式如下：



其中，注意，在这种情况下，ML estimate和最小化reprojection error function是相同的。

**Mahalanobis distance：**在一般的gaussian案例中，一个假设是向量X满足具有covariance matrix 的gaussian distribution function。上述情况等价于具有identity multiple的covariance matrix。

最大化log-likelihood等价于最小化mahalanobis distance



在每个图像中都有错误的情况下，并假设一幅图像中的错误与另一幅图像中的错误无关，则appropriate cost function公式如下：



其中和是两张图片中measurement的covariance matrix。

最后，如果我们假设所有点和点的误差都是独立的，且各自对应着covariance matrices和，则上式可以进一步写为：



这个方程允许加入anisotropic covariance matrix，以两条non-prependicular line的交点来计算point的位置。在两幅图像中的一个点完全已知的情况下，错误仅限于另一个图像，这样（4.16）中的两个求和项中的一个将消失。

## 4.4 Transformation invariance and normalization

现在我们开始讨论第4.1节中的DLT算法的性质和性能，以及讨论如何将DLT算法与最小化geometric error相比较。第一个主题是算法对图像中不同坐标选择的不变性，很明显，一个算法的结果依赖于图像中坐标系的原点、尺度、甚至方向等任意选择，通常是不可取的。

### 4.4.1 invariance to image coordinate transformation

图像坐标原点有时会给在图像，有时会出现在图片中心。这是否会对计算transformation的结果产生影响。如果image coordinate乘以某个倍数，则算法的结果是否也有可能发生变化？更常见的问题是，通过最小化cost function来计算homography的算法多大程度上依赖于image中的coordinate选择？在运行算法之前，Image coordinate可以通过similarity，affine或者其他projective transformation改变。以上这些会实质性地改变这个结果吗？

假设，一张图片中的coordinate x 被替代，并且另一张图片中的coordinate 被替代，其中和是3\*3的homography。将这两个式子代入到等式中，可以推导出。这种关系意味着是的transformation matrix。因此，提出如下的另一个计算到transformation的方法：

1. 根据transformation和来变换image coordinate
2. 通过correspondence找到transformation
3. 令

用这种方式找到的transformation matrix H适用于original untransformed point correspondence。对于transformation和应该做出什么选择，现在还没有说明，现在要决定的问题是确定算法的输出是否与所应用的transformation和无关。理想情况下，它应该是这样的，当transformation和是similarity transformation时，由于在图像中选择不同的尺度、方向或坐标原点不会对算法的结果产生实质性的影响。

在接下来的章节中，最小化geometric error是不变的similarity transformation。在另一方面，对于第4.1节中所述的DLT算法，solution在similarity transformation中不会改变。在用DLT算法求solution之前需要对输入数据进行归一化。这种归一化将消除在图片中任意选择origin和scale所造成的影响，并且归一化也意味着combined algorithm在图片的similarity transformation中是不变的。稍后将讨论适当的规范化转换。

### 4.4.2 DLT的非不变性

是一系列correspondence，matrix H是将correspondence应用到DLT算法后计算出的结果。进一步考虑correspondence，其中，，并令。在第4.4.1节之后，这里需要决定的问题如下：

1. 将correspondence代入到DLT算法能否计算出transformation 

我们将使用以下符号：matrix从correspondence推导出的DLT equation matrix，且A是有叠加形成的2n\*9的matrix。类似的也是由correspondence定义，其中，。

令为similarity transformation乘以标量s的结果，而为任意的projective transformation。进一步，令为任意2D homography，令，，其中和是和的向量。

证明：向量，请注意，是由的前两个项组成的向量。类似的，进而推导：



其中，表示的辅助因子矩阵，对于一个一般性的transformation T，error vector 和（即和的前两个组成部分）并不是简单地相关的。当为 similarity transformation时，，R时rotation matrix ，t是translation，s是scaling factor。此时，将仅应用于的前两个组件，可以得到：



因为旋转不影响vector norm，所以，按需要这个结果可以用代数误差表示为：



在和之间有一个一一对应的correspondence，因此和的最小化代数误差可以通过公式联系起来，并且，H可以通过得到。然而，这个结论是错误的。虽然和这样定义会产生相同的误差，已知条件，作为约束施加的解，并不等同于条件。因为和并不是简单的关联关系。所以在和之间不会因为一一对应关系而导致相同的错误，只是需要满足constraint 



这样，这个transformation方法计算出了不同的transformation matrix。这是DLT算法的一个相当不受欢迎的特性，因为结果随着坐标的改变而改变了，哪怕是仅仅是对坐标原点的改变。如果范数最小化的约束在变换下是不变的，那么和就可以通过正确的方式联系起来。

### 4.4.3 geometric error的不变性

本节将介绍在similarity transformation下通过最小化geometric error找到的H具有不变性。像前几节一样，给定correspondence和transformation matrix ，以及correspondence，其中，，令。假设和表示P2中的euclidean transformation。可以推导公式如下：



最后一个等式成立是因为euclidean distance在euclidean transformation下不变的。这说明如果H使geometric error最小，则使变化后的geometric error最小化，因此，最小化geometric error在euclidean transformation下是不变的。

对于similarity transformation，geometric error乘以scale factor，这样可以像euclidean transformation 里的最小化那样最小小transformation correspond。最小化geometric error对similarity transformation是invariant的。

### 4.4.4 归一化transformation

正如4.4.2中展示的，使用DLT算法计算2D homograph依赖于坐标系的选取。在图片的similarity transformation中这个结果是有可能改变的。这意味着计算2D homography过程中一些坐标系是优于其他坐标系的。本节描述了一种normalization 数据的方法，包括image coordinate的translation 和scaling。在使用DLT算法之前需要进行这种normalization，最终得到的H相对于original coordinate system 会有一个适当的修正。

Normalization除了可以提高H的精度，还可以使algorithm面对任意选择的scale和coordinate origin时是保持不变的。因为normalization通过选择一个canonical coordinate frame的方法抵消了coordinate改变产生的影响。所以，algebraic minimization是在一个福鼎的canonical frame中进行的，并且DLT算法在similarity transformation中是不变的。

Isotropic scaling（各项同性缩放）normalization的第一步是translate每个图片的coordinate，目的是将所有点的centroid带到原点上。同时coordinate也被缩放了，即中x轴，y轴，z轴都缩放了相同的量级。最终我们选择缩放的量级将使这些点到原点的平均距离等于。

这意味average point等同于

上面所有内容总结如下：

1. 这些点被平移，使它们的质心在原点处
2. 然后对这些点进行缩放，使这些点到原点的平均距离等于。
3. 这种转换分别应用于这两幅图像。

**为什么normalization 很重要？**在4.2节中给出了推荐的带有normalization版本的DLT算法。我们现在好奇为什么包含data normalization的DLT算法为什么比4.1的DLT算法更值得被推荐使用。

4.1中的DLT算法对使用SVD来得到的解。实际上没有精确解，但是有V得到的vector h，可以通过最小化来得到一个解。这等同于在frobenius norm中通过找到最接近于A的秩为8的矩阵,得到的h是的精确解。中，当smallest value设为0时，就是，矩阵的秩为8，我们需要尽量减少frobenius norm中与A的差异，因为：

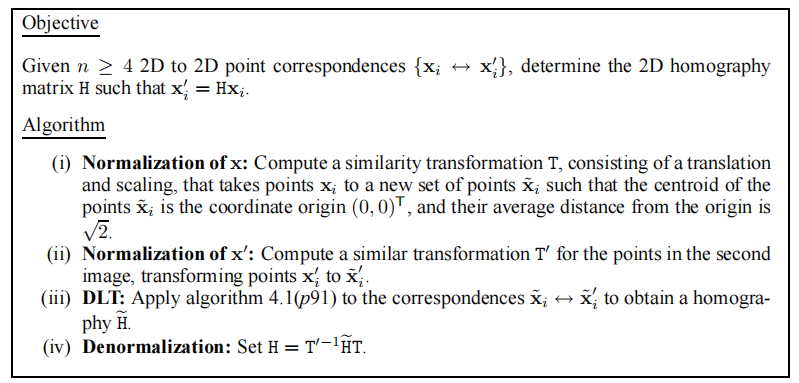


其中就是frobenius norm，例如所有项平方和的平方根。

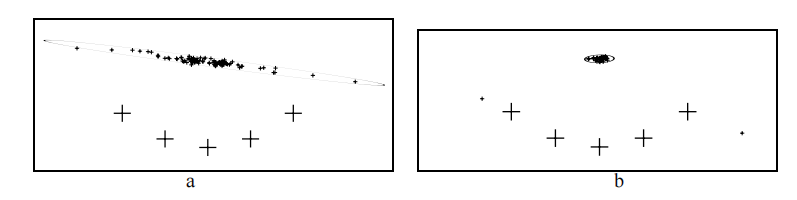
如果图片中的，没有normalization，假定坐标，x,y比w要大很多，在A中将为量级，而只有量级，直接为unity。用Aˆ替换A意味着一些entries增加，另一些entries减少，通过这些改变可以使平方和最小化。然而，问题在于，将扩大100倍意味着图像点发生巨大的变化，而将扩大100倍只意味着轻微的变化。所以这就是A中所有entries需要normalization的原因。

归一化的效果与DLT方程组的条件数有关，或者更准确地说，与方程矩阵A的第一奇异值与第二奇异值的比值有关。使用了normalization transformation后对于精确数据和infinite precision arithmetic来说得到的结果是独立的。然而，在存在噪声的情况下，该解将偏离正确的结果，且条件处越多，偏离的程度越大。

这种数据归一化对DLT算法结果的影响如图4.4所示。这里得出的结论是，数据归一化会有更好的结果。Data normalization是DLT算法中非常重要的一个步骤，并不是可有可无的一个选项。

4.2节中建议的加入data normalization的DLT算法。

用5个点来计算2D homography，这5个点都被映射到相同的coordinate中，这样homography H也可以被唯一映射。现在，100次试验，每个点受到0.1像素高斯噪声。通过DLT算法计算出的H将一个点从第一张图映射到第二张图中，用小叉符号表示这一点的100个投影，并给出了由其散点矩阵计算出的95%椭圆。.(a)是没有数据归一化的结果，(b)是加了数据归一化的结果。



**Non-isotropic scaling（无同向性缩放）：**另一个方法是在无同向性缩放中，点的质心像前面一样被平移到原点，然后，这些点形成了一个关于原点的点云。然后进行缩放，直到点云的两个主成分等同于单位向量。这样，点云将形成一个围绕原点半径为1的近似对称的圆云。在[Hartley-97c]中给出的实验结果表明，无同向性缩放所需的额外努力并不会求出比各向同性标度明显更好的结果。

**使用接近无穷大点进行缩放：**有一种情况是从一张图片和一个无限平面来估计homography。如果观察方向足够倾斜，那么在图像中就可以看到平面上非常遥远的点——前提是视界是可见的，即使是无穷远处的点（消失点）。在这种情况下，在原点处设置质心来normalize无限平面上的点的坐标是没有意义的，因为质心可能有非常大的坐标，或者没有定义。

## 4.5 interative minimization methods 迭代最小化方法

本节描述各种cost function。最小化这些cost function需要迭代，但是，迭代技术与线性算法相比，往往有一些缺点：

1. 迭代更慢
2. 它们通常需要一个初始估计来开始迭代。
3. 它们有不收敛的风险，或者不收敛到一个局部最小值，而不是一个全局最小值。
4. 什么时候停止迭代是一个很棘手的问题

因此，迭代技术通常需要更仔细的实现。

迭代最小化技术一般包括五个步骤：

1. cost function：先选择一个cost function
2. Parameterization：要计算的transformation需要有限数量的参数表示。并不必要有一个最小集合的参数，事实上，过度参数化往往也有好处
3. Function specification：使用参数集来表示cost function。
4. Initialization：计算一个合适initial parameter estimate。通常使用线性算法来计算initial parameter estimate。
5. Iteration：从initial solution开始，迭代细化参数，以最小化代价函数。

**A word about parametrization：**对于一个给定的cost function，通常有几种参数化的选择。参数化的一般方法是选择一系列parameter来覆盖需要最小化的complete space，这样可以很方便的计算cost function。例如，H可以由9个参数参数化——也就是说，它被过度参数化了，因为实际上只有8个自由度，过度参数化是没有意义的。最后需要最小的参数化（即自由度）将只涉及8个参数。但是过度参数化也不会带来什么太大坏处，只要选中的参数是需求的类型。尤其是对于homogeneous object，比如3\*3的矩阵，其实并不推荐使用remove scale-factor ambiguity的方法来最小化参数。因为，一个良好的优秀的非线性最小化算法自然会便面向冗余的方向最小化计算。当minimal parametrization使用时，cost function surface会变得更加复杂，这样，陷入局部最小的可能性就更大了。

在选择parametrization时容易遇到的另一个麻烦是在将transformation限制到一个特定类时。例如，已知H是homology，则H可以被参数化为：



其中是标量，v是3维向量，Homology有5个自由度。如果H通过矩阵的9个元素参数化，则估计出来的H不太可能是一个homology。如果H用µ、v和a（总共7个参数）参数化，那么估计出来的H保证是homology。

#### Function specification

在4.2.7中估计问题的一般都关注包含model surface S的测量空间。给定一个x,来估计位于S上接近于x的点。在non-isotropic gaussian error distribution下，用mahalanobis distance来表示最近距离。Iterative minimization 方法将用来估计模型，通过参数拟合来迭代估计，model surface S被局部参数化，随着这些参数变化来最小化measured point的最小距离。

1. 一个有一个带有协方差矩阵Σ的测量向量X∈IRN
2. 参数用向量表示
3. 定义映射
4. 使用squared mahalanobis distance作为cost function



实际上，我们试图找到一组参数P，使其在mahalanobis distance下有f (P) = X，使f (P)尽可能接近X。当要最小化的代价函数是mahalanobis distance时，levebverg-marqurdt算法是迭代最小化的一般工具。现在，我们将展示本章中描述的各种不同类型的cost function。

**Error in one image：**确定第一个图像中的点xi的坐标，随着H的改变来最小化cost function。



Measure vector X由点xi的2n个非齐次坐标组成，H是homograph，h是H的entries，Hxi表示非齐次坐标。function f可以定义为：



**Symmetric transfer error：**在symmetric cost function中：



Measure vector X由xi的非齐次坐标和点的非齐次坐标组成的4n向量，function f可以定义为：



Reprojection error：minimizing cost function的问题点在于需要对所有的和transformation matrix H进行最小化。如果有许多point correspondence，则计算量会很大。所以急需要将和H参数化，参数化后有2n+9个参数。Measure vector 包含所有xi和的非齐次坐标。



**Sampson approximation（桑普森近似）：**与包含2n+9个parameter的reprojection error相比，最小化one image error和symmetric transfer error只需要9个参数（H中的9个entries）----这就简单多了，而Sampson approximation也只涉及9个参数。参数的个数是很重要的，因为使用levenberg-marguardt算法计算包含m个参数的非线性最小化问题时涉及m\*m个线性等式，其复杂度为，因此需要控制参数的个数。

桑普森误差避免了重投影误差的2n+9个参数的最小化，最小化只需要h的9个参数。在实践中，如果误差与测量结果相比很小，这种近似就给出了很好的结果。

#### Initialization（初始化）

参数的初始化可以通过线性方法完成，例如，4.2节中的DLT算法可以通过迭代最小化找到包含9 vector h的H。一般来说，如果有n个≥4对应，则所有对应都将在线性解中使用。另一个方法是在参数空间进行采样，从每个采样的起点进行迭代，并保留最佳结果，这种方法只有在参数个数很少的时候才可以使用。另一种初始化方法是完全不需要任何有效的初始化，仅仅需要在参数空间给定一个固定的点，并从这个点开始迭代即可；但是这个方法并不适用所有情况，因为迭代可能会陷入局部最小，即使是最好的情况，迭代次数也会随着最终的结果计算而增加。因此，使用一个良好的初始化方法是一个十分重要的课题。

#### Iteration methods（迭代方法）

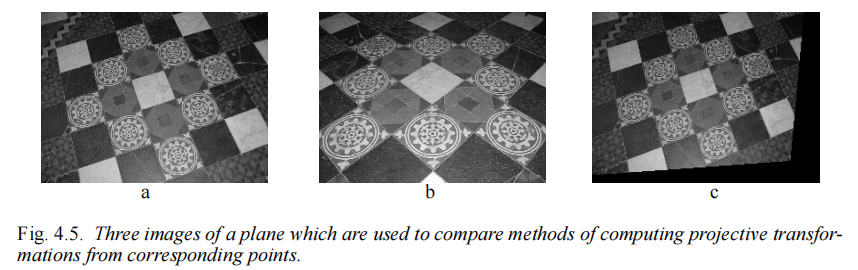
迭代完成最小化的方法有许多，最常用的就是levenberg-marquardt，另一常用的方法是powell方法

#### Summary（总结）

本节的所有内容实际上在4.3节中都有收集，4.3节中gold standard和Sampson 方法用来估计两幅图片中homography mapping。

## 4.6 experimental comparison of the algorithm

Fig4.5中的第一对图片（a）和（b）中有55个point correspondence。所有的方法计算这两个图片得到的效果都是一样的（除了affine method）。Optimal residual大于所得到的结果，因为噪声干扰水平低于一个pixel。Fig4.5中的（c）是通过(a)重采样进行合成的，第二对由(a)和(c)组成，这组有20个point correspondence。在这种情况下，几乎所有方法几乎是最优的，如表4.1所示。例外是affine method 和unnormalized linear method，因为（a）不是affine 变换所以affine method表现不理想，而为什么非标准化的方法表现会很糟糕（尽管可能没有那么糟糕）。为什么unnormalized linear method在第一对表现好，在第二对表现糟糕，目前还不清楚原因。



进一步的评估结果在fig4.6中。要评估的transformation是棋盘图片（a）映射到与直角坐标轴对齐的正方形网格。正如图上看到的，图片是失真的。实验中，在图片中随机选择点并与square grid上的点对应起来形成correspond point。DLT算法和gold standard algorithm在minimum error和residual error这两个指标上进行比较。注意，当只有5个pixel的噪音时，DLT算法的表现更好。然而，当噪音有10个pixel时，DLT将无法正常计算，因为对于一个200pixel的图片来说，10pixel的噪音过于高了。

## 4.7 robust estimation

Correspond point是error的来源，因为correspond point服从高斯分布。不匹配的点是高斯误差分布的离群值。因此需要从correspond point中剔除不匹配的点（outliers），用匹配的点（inliers）来计算出homography。包含这一步的计算方法被称为robust estimation。

### 4.7.1 RANSAC

我们以一个简单的例子开始——根据一群二维点估计直线。问题是在图4.7a中可以看到：给定一系列2D points，找到一条直线，是的所有点到直线的square perpendicular distance最小，并且同时满足该直线上的有效点个数超过t。这实际上可以被分解为两个问题：首先是用数据拟合直线，然后将原始数据分为inliers和outliers。t值的设定是根据衡量noise的多少得到的。目前已经有许多的robust algorithm，这些算法在一定的程度上都依赖于就算outlier的比例。例如，如果已知只有一个outlier，然后，每个点可以依次删除，并从其余部分估计直线。这一节中我们主要讨论一个非常成功的robust algorithm——the random sample consensus（RANSAC）。RANSAC算法能够处理很大比例的outliers。

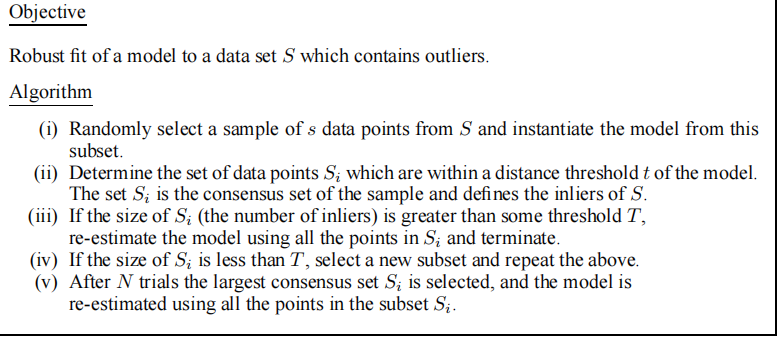
RANSAC的思路非常简单：首先随机选取两个点确定一条直线；计算点到直线的距离，满足距离阈值的点就是这条直线的support；重复随机选择一定次数之后，最大support可以被计算出来；这些满足距离阈值的support就是inliers，直观上看如果support中包含了outlier则，这个直线应该是不可能获得足够的support的，如图4.7b中所示。

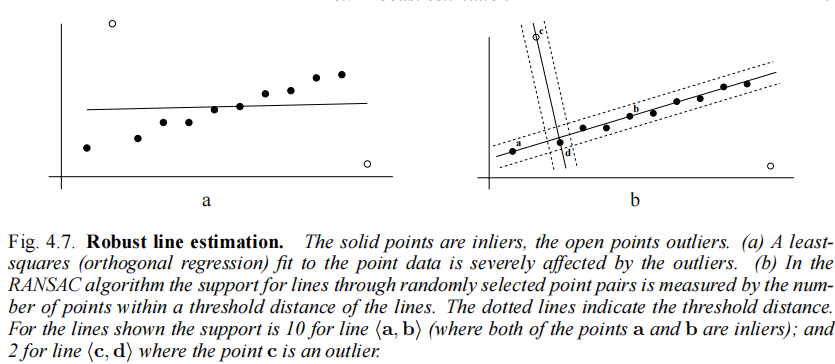
更进一步，计算一个直线的分数在得到最优拟合结果是有优势的。例如在图4.7b中，直线<a,b>有10个support，而直线<a,d>只有4个support。即使这两个直线的support中都不包含outliers，最终也会选中直线<a,b>。

更一般化的，我们希望拟合一个model。随机选择的样本是有原始数据的最小子集构成，以直线为例，随机选择的样本就是两个point。当model不是直线而是一个planar homograph，原始数据是2D point correspondence，随机样本就由4个correspondence。下面章节将描述用RANSAC估计homograph。

与传统的smoothing technique不同，RANSAC不是使用尽可能多的原始数据来获得初始解，然后试图消除无效的数据点；RANSAC使用尽可能小的初始数据集，然后尽可能使用一致的数据扩展此数据集

RANSAC步骤

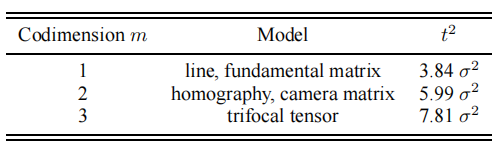




1. **什么是距离阈值？**我们通过确定距离阈值t来得到α概率的point为inliers。这个计算需求知道模型中inliers的距离分布。一般来说距离阈值是依据经验来设定的。但是如果我们知道了高斯分布是服从zero mean和standard deviation ，那么距离阈值t也可以通过计算得到。表示point distance的平方和，表示所服从的高斯分布，其中m表示自由度，m等于model的codimension。Line的codimension为1——因为只需要计算point到line的垂直距离。如果一个model的codimension为2，距离的平方是x和y测量误差的平方和。Cumulative distribution公式表示如下：



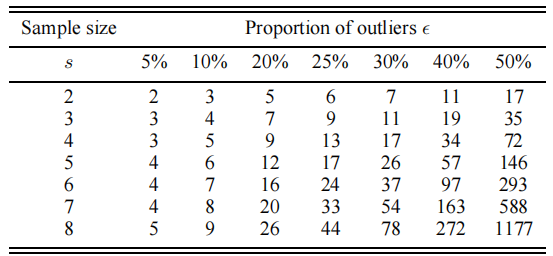
其中为0.95，表示point中有95%的点为inliers。这意味着有5%的次数中inliers会被拒绝。在为0.95时t的值如下表所示



1. **需要采样多少次？**N表示采样N以此确保有p的概率随机采样的s个point中至少有一个不是outlier。通常p的值为0.99，w表示任意选一个point都是inliers的概率，则表示outlier的概率。得到公式：

 （4.18）

给定，s和p=0.99得到对应的N如下表所示





例子4.4：如图4.7中一共有12个point，其中有两个outlier，所以。S=2,查表4.3可以得到N=5，即至少采样5次。我们可以将5次与遍历所有point中的point对所需的遍历次数做对比，遍历所有的point对需要次（排列组合）

Note：（1）采样的次数与比例相关，而与outlier的数量无关，采样的数据需要小于outlier的数量。因此，即使当outlier的数量很大，采样的计算成本也是可以接受的。

1. sample的次数随着minimal subset的增大而增大。当提高minimal subset的数量时，是有一定益处的，例如，用3个point或者4个point你个一条直线可以得到一个更好的estimate结果。然而不能无脑提高，需要和computational cost做一个平衡。
2. **多大的consensus set**是可以被接受的？一个经验法则是，如果consensus set与inliers的数量类似，也被认为是给定了outlier比例后计算出的data set。例如4.7中，n个data point中，，则可以计算出。
3. **自适应地确定样本的数量。**大多数情况下的值是未知的，一般都是按照最差的情况来初始化，然后随着找到consistent set来更新值。例如，按照最坏的情况将令，随着计算的进行，发现consensus set占原始数据的80%，则将的值从0.5更新为0.2。这种通过consensus set“探测”数据的想法可以重复应用，以自适应地确定sample N的数量。继续上面的例子，是从最坏情况估计决定了初始N。当consensus set的比例超过50%时，inliers的比例至少是50%。根据公式（4.18）可以更新的值，在每个样本上重复这个更新，当低于当前估计值的consensus set时，N的值将再次减少。一旦计算出N，算法就会终止。一个可能的情况是有确定的采样次数N小于sample的数量。在这种情况下，已经执行了足够的样本，并且算法终止。这种自适应的方法非常有效，的初始值可以被定为1，此时采样次数N的初始值是infinite。明智的做法是使用一个保守的概率，令=0.99。

### 4.7.2 robust 极大似然估计

RANSAC算法是计算出inliers和outlier，同时也提供了对模型的估计值M0，RANSAC的最后一步是利用inliers重新估计模型，这个重新估计的过程涉及到ML cost function。以line为例，ML estimation等价于orthogonal regression，并且会得到一个closed form solution。一般来说ML estimation涉及iterative minimization，和minimal set 估计。

RANSAC只有一个缺点，就是inliers和outlier的分类是不能被改变的。在model拟合出最佳的consensus set后，当距离阈值被应用到新的model时，可能会出现额外的point需要被归类为inliers的情况。例如，图4.8中，line<A,B>通过RANSAC被拟合出来，这条线有4个point作为一个support，他们都是内点。对这四点进行最优拟合后，又出现了10点需求被分类为inliers。要解决这个问题就需要进行两个步骤：最佳拟合得到inliers，根据（4.1.7）重新分类inliers；然后不断迭代知道所有inliers都被覆盖到。在这个阶段，通常使用根据整数到模型的距离加权的最小二乘拟合。

Robust 损失函数：表示需要最小化的内容，引出cost function：



其中表示point error，表示robust cost function其中outlier有一个固定的cost。表示（4.17）中的threshold，和的定义一致。Inlier的quadratic cost来自于Gaussian error model；robust cost function中outlier的constant cost来自于假设outlier遵循扩散分布或均匀分布，它的对数似然值是一个常数。Outlier可以通过cost function中的对的threshold来剔除。阈值化本身的问题是，它将只导致包含outlier，因为它们不会产生任何成本。

Cost function D将最小化所有point（包括outlier和inlier），D和的区别仅在于一个常数（由outlier数量的4倍给出）。然而，随着最小化的进展，outlier可以被重新指定为inlier。

### 4.7.3 其他robust算法

Least median of square estimation（LMS）与RANSAC不同的点在于，RANSAC是根据满足距离阈值的point的个数来打分，而LMS则是根绝所有point的距离中位数来打分。LMS的优势是不需要设置阈值或对误差方差的先验知识。缺点是如果超过一半的数据是离群值，那么距离中位数也将是一个离群值。解决方案是使用异常值的比例来确定选择距离。例如，如果有50%的异常值，那么就应该选择低于距离中位数的那个值。

LMS和RANSAC的共同点是都要能够处理很大比例的outlier。但是但outlier比例很小的时候，使用其他的robust算法效果会更好。这下算法包括case delete（其中每个点依次被删除，将模型与剩余数据进行拟合）； iterative weighted least-squares（迭代加权最小二乘，其中数据点对拟合的影响与其残差成反比）

下表详细描述加入RANSAC后计算两张图片之间的homography的步骤

使用RANSAC自动估计两幅图像之间的homography。

1. interest point：计算出每张图片中的interest point
2. Putaive correspondence：基于其强邻近区域的proximity和similarity计算出matched interest point
3. RANSAC robust estimations：假设基于上面步骤采样到N个样本
4. 随机选择4个correspondence 计算出homography H
5. 计算出每对correspondence之间的距离
6. 通过计算出H对应的inliers

选择inliers数量最多的H。

1. optimal estimation：通过被标记为inliers的correspondence重新估计H，估计方法是使用levenberg-marquardt算法来最小化ML cost function。
2. Guided matching：进一步的interest point correspondence可以使用被估计出的H定义个一个关于tansferred point位置的搜索区域得到。

最后两步可以进行迭代，直到对应项的数量稳定为止。

## 4.8 自动计算homography

算法输入是image，输出是homography。

第一步，我们需要确定是先有鸡还是先有蛋的问题——一旦确定了确定interest point，就可以使用robust estimation（例如RANSAC）确定homography。反过来也是，一旦确定了homography就可以确定interest point。

**确定假定的corresponddence：**在未知homography的情况下，假定correspondence的目的是要提供一个初始的correspondence point。这些correspondence的大部分应该是正确的，但是不要求所有的correspondence都完美匹配，因为后面RANSAC将消除那些不匹配的correspondence。这些putative（假定）correspondence是通过探测每张图片中使用强度邻域的接近性和相似性的interest point得到的。为了简洁起见，interest point将被称为“corners”。然而，这些corners不一定是场景中的物理角落的图像。这些corner由图像的auto-correlation function的最小值来定义。

对于image1中的每个corner（x,y）在以（x，y）为中心的正方形搜索区域内选择图像2中最高的匹配点。但有时候会出现冲突，image1中的一个点可能会在image2中找到多个匹配度高的点。这个时候只选择匹配度最高的那个点。使用square sum of intensity differences（SSD）代替cross-correlation（CC）来计算similarity measure。CC对affine mapping的intensity是不变的，而SSD对affine mapping并不是不变的。当图像之间的intensity变化很小时，SSD通常是首选的，因为它比CC更敏感而且计算上更便宜。

**Homography中的RANSAC：**将RANSAC应用到putative correspondence set中来计算homography。样本的最小大小为4，因为4个correspondence可以计算出一个homography。但是真正使用的样本大小还是要根据outlier的比例决定。

现在需要考虑两个问题：1.distance设置为多少合适？2.如何选择样本？

1. distance measure：评估homography误差最简单的方法就是使用symmetric transfer error，例如：，是point correspondence。一个更好的，但更昂贵的评估方法是reprojection error，其中表示完美的correspondence，这个度量更昂贵，因为也必须计算。第三个评估方法是sampson error。
2. Sample selection：这里有两个问题。首先，degenerate sample应该被忽略（例如，如果四个点中的三个是共线的，则不可能计算出homography）。第二，样本应该由在图像上具有良好空间分布的点组成——这是因为extrapolate problem，homography将准确地绘制出由计算点所横跨的区域，但是当距离超出这个范围时准确率将下降（例如图片的上下左右四个角）。

Robust ML estimation 和引导匹配：最后一个阶段的目的有两个方面：（1）利用inliers 来估计homography。（2）计算出homography后，可以从putative correspondencep set中获得更多的嵌入匹配。然后，通过最小化ML代价函数和inlier计算出改进的homography。这一个阶段可以通过两种方式实现：一种方法是在inliers上进行ML估计，然后使用新的H重新计算inliers，重复这个循环，直到inliers的数量收敛。ML cost function minimization使用的是A6.2（p600）节中描述的 Levenberg–Marquardt algorithm。另一种ML方法是使用4.7.2节中提到的robust ML cost function——同时计算出homography和inliers，这个方法的缺点是在代价函数最小化时所产生的计算量。因此，循环方法通常更有吸引力。

### 4.8.1 application domain(应用领域)

该算法要求能够在图像上相当均匀地重建interest point，而这又需求一定的场景和分辨率。

搜索窗口邻近约束对视图之间的角点（视差）的图像运动设置了上限。 然而，如果不应用此约束，该算法也不会失败，并且在实践中，邻近约束的主要作用是降低计算复杂性，因为较小的搜索窗口意味着必须评估的角匹配较少。

最终，该算法的范围受到角邻域相似性度量（SSD 或 CC）在消除对应关系方面的成功的限制。 失败通常是由于缺乏空间不变性造成的：这些度量仅对图像平移不变，并且会因变换而严重退化，在此类之外，例如图像旋转或图像之间的透视显着差异。 一种解决方案是使用对图像之间的单应性映射具有更大不变性的测量，例如旋转变化的测量。

### 4.8.2 实现和细节

Interest point可以通过棋盘格获取。邻域相似性度量的阈值是故意设置的比较保守的，目的是为了最小化不正确的匹配。

对于例子图4.9，图像为640×480像素，搜索窗口为±320像素，即整个图像。当然，考虑到actual point 的差异，也可以使用更小的搜索窗口。通常在video sequence中，一个±40像素的搜索窗口就足够了（即一个以当前位置为中心的80边的正方形）。初始阈值为t = 1.25像素。最后一共需要43个样本，采样的过程可以见表4.4。Guided match需要 MLE-inlierc classification循环的两次迭代。d⊥像素误差的RMS value在MLE之前为0.23，在MLE之后为0.19。最后levenberg-marquardt需要迭代10次。

## 4.9 closure

本章将说明用于估计代表multiple view relation的tensor的问题和技术。这些在书的每个计算章节中也重复出现。每次出现时都需要最少的correspondence required；需要避免degenerate configuration；也需要最小化代数和几何误差；也需要对tensor施加内部约束的参数化。

### 4.9.1 相关文献

DLT算法至少可以追溯到 Sutherland。 Sampson则发表关于圆锥形拟合的经典论文，Hartley在计算机视觉文献中公开了标准化。关于数值方法的相关阅读可以在Numerical

Recipes in C，以及Gill和Murray [Gill-78]提出的迭代最小化中找到。Fischler and Bolles的RANSAC是最早的鲁棒算法之一，RANSAC实际上是为了解决计算机视觉问题（pose from

3 points）而开发的。原论文的论证非常清楚，非常值得一读。其他关于robustness的背景材料可以在Rousseeuw的书中找到。Robust估计在计算机视觉中的主要应用是estimating fundamental matrix（第11章），Torr和Murray 使用RANSAC robust 估计，Zhang等人使用LMS robust 估计。由Torr and Zisserman描述了一个自动ML估计来计算homography。

### 4.9.2 注意

（1）**计算homographies ：**（4.1-p89）和（4.3-p89）假定的维度是三维的，从而可以定义一个cross-product。然而，（4.3）可以使用推广到所有维度的方式来求导。假设= 1，我们可以通过计算位置sacle factor。从第三个坐标我们可以得到，并将其代入原方程得到：



**（2）通过ideal point计算homograph：**如果其中一个点是一个ideal point，那么=0则有等式（4.3）可以转化为一个等式（尽管（4.1）中包含两个独立等式）。为了避免这种简化方式，同时又想包含最少数量的方程，一个好的方法如下。我们换种方法写公式，写成如下形式：



其中是与行正交的矩阵，所以有。的每一行都可以得到一个关于H的单独的线性方程。矩阵可以通过删除满足的正交矩阵M的第一行来得到。Householder matrix是一个容易构造的具有期望属性的矩阵。

1. **缩放 unbounded point set：**在平面上的point在无穷远或接近无穷远的情况下，使用本章提出的isotropic（或non-isotropic）sacling schemes对坐标进行归一化既不合理也不可行，因为质心和尺度是无限的或接近无穷远的。一个更好的方法是对这样的point set进行归一化：



其中和对应各homograph coordinate，而这些条件不再意味着质心是在原点处。

**（4）DLT的transformation不变性**：我们通过包含各种约束来最小化代数误差，证明如下：

1. 如果在的约束下最小化，最后得到的结果在change scale下具有不变性。
2. 如果在的约束下最小化，最后得到的结果在similarity下是不变的，
3. 如果在的约束下最小化，最后得到的结果在affine下是不变的。

**（5）图像坐标导数的表达式**：对于map ，推导出一下表达式：

1. 对x求偏导

其中是H的第j行

1. 对H求偏导

**（6）具有non-isotropic error分布的Sampson error：**在第4.2.6节中，对Sampson误差的推导假定点是用circular error distribution来测量的。在使用covariance 矩阵测量Point 时，适合使用minimize mahalanobis norm。在这种情况下，与（4.11-p99）和（4.12-p99）对应的公式为：





注意，如果两个图像中的测量值是独立的，那么covariance 矩阵将是两个对应图片的2×2的对角矩阵。

**（7）Sampson error 编程提示：**在2D homograph估计问题中，事实上，这本书中考虑了其他类似的问题，x坐标的损失函数是multilinear的。这意味着偏导数可以非常简单地计算出来。计算如下：



以上计算的结果是精确的，而不是近似值。这意味着为了编程目的，不需要编写一个特殊的例程来求导——计算 routine就足够了。用表示第i位包含1的向量，否则为0，可以得到

进一步推导得到：



还要注意的是，在计算上，直接使用求解更加有效，而不是使用求解。

1. **在affine transformation中求解最小几何误差**：给定correspondence ，找到一个使几何误差最小化的仿射变换。下面我们将逐步推导一个基于samposon approximation的线性算法，这在这种情况下计算出来的结果是精确的。完整的方法总结在算法4.7中。
2. 结果表明最优仿射变换将的质心到的质心，因此，通过将point 转换到质心，确定变换的translation部分。只需要确定的左上角2×2矩阵部分，它表示线性translation部分。
3. 当且仅当时，点位于上。所以是R4的一个二维子空间。
4. 任何合适的H2二维子空间都可以表示为。给定，measurement ,则estimation task变为最佳拟合的codimension 二维子空间。
5. 给定一个行为的矩阵M，的最佳拟合子空间可以通过M的两个最大奇异值对应的奇异向量V1和V2组成。
6. 通过解方程，得到了与V1和V2所张成的子空间对应的

三维世界中从line correspondence中计算出homograph：假设我们只能从line correspondence中计算出4\*4的homograph。有两个问题：1.需要多少个correspondence？2.如何公式化代数约束来得到H的解？可能会先考虑使用4个line correspondence，因为在三维中每个line会有4个自由度，这样4个correspondence可以提供4\*4=16个约束，从而满足自由度为15的H。然而4个line在计算transformation时会遇到degenerate问题，因为2D isotropy subgroup的存在。这个问题将在后面讨论，H中的线性方程可以用以下方法得到；



其中H可以将由point（x1，x2）确定的line转换到由plane（π1，π2）确定的intersection上。该方法是在[Oskarsson-02]中推导出的，其中需要找到更多的细节。

# 算法评估和错误分析---algorithm evaluation and error analysis

本节表述了评估算法质量的方法，通常对一个variance或者一个transformation估计是不够的，measure的confidence或者uncertain也是有必要的。本节将介绍两个方法来计算uncertain。第一个方法是基于linear approximation并且涉及到连接各种雅可比矩阵表达式。第二个方法是Monte Carlo method.

## 5.1性能的界限--bounds on performance

一旦开发了一种算法来估计某种类型的变换，就该测试其性能了。这可以通过真实或合成数据来测试。在本节中，将考虑使用合成数据的测试，并概述一种测试方法。

我们回顾下notation convention：

* x表示一个被测量的图像点。
* 使用或者表示estimated结果
* 使用或者表示真实值

通常，测试将从两个图像之间的一组correspondence 开始。这样的correspondence的数量会很多。通过给定的fixed projective transformation选择Correspondence point。

接下来，将通过zero-mean guassian random variable在图像测量的x坐标和y坐标中中加入人工高斯噪声。加入噪声后的point用和表示。然后运行估计算法来计算估计的量。Estimation algorithm然后计算estimated quantity。对于章节4中的2D projective transformation problem，这意味着projective transformation也可能估计正确的原始无噪声图像点。然后，根据计算出的模型与（有噪声的）输入数据的匹配程度来评估该算法，计算estimated model和具有噪音的原始数据的差距有多近。该过程应该在不同的噪声下进行多次（虽然每次都有相同的噪声差异，但是random number generator 生成的seed也不同，），以获得统计上有意义的性能评估。

### 5.1.1 image中的error

我们继续研究2D homography estimation问题。为了简单起见，我们考虑只将噪声添加到第二个图像的坐标中的情况。对于所有的i 都有。设是两幅图像之间的噪声匹配点，噪声通过往一系列完美匹配的数据中注入方差为的高斯滤波生成。设有n个这样的匹配点。从这些数据中，使用第4章中描述的任何一种算法来估计一个射影变换。显然，由于在的坐标中注入了噪声，估计出的通常不会精确地将到和映射起来，也不会精确的将i和映射起来。The RMS (root-meansquared) residual error表达式如下：



测量noisy input data（）和estimated point 之间的平均差值。也被称为残差。它度量transformation match与input data的匹配程度，因此它是estimation procedure中的合适quality measure。

Residual error的值本身并不是对所得到解的质量的absolute measure。例如，2D projectivity problem中输入数据由4 match point。因为projective transformation是由4个独立的point correspondence，随便什么算法都可以通过这些match point和计算出。这意味着残差为零。已经很难再找到比这个算法更好的选择。

请注意，将projective point与input data匹配，而不是与将将projective point与原始的无噪声数据匹配起来。事实上，由于noise free和noise coordinate之间的方差为，在有4个point的情况下，projective point和noise-free data之间的方差也为。因此，在4个point的情况下，model完美地拟合了noisy input data（即残差为零），但没有给出一个非常接近于true noise-free value。

当使用超过了4个point matches，残差值会增加。事实上，随着测量值（匹配点）数量的增加，estimated model应该越来越接近于无噪声的真实值（noise-free true value）。渐近地，方差应该与点匹配的数量成反比地减少。同时，残差也会增加。类似的，方差应该与point match的数量成反比。

### 5.1.2 error in both image

在两幅图像有误差的情况下，残差为



其中，和是的关系。

### 5.1.3 optimal estimator（MLE）

在general framework中将考虑estimation performance的bounds，然后专门用于一种或两种图像中的两种错误情况。目的是推导出最大似然估计（MLE）的expected residual error。如前所述，几何误差的最小化等价于MLE（maximum likelihood estimate），所以任何实现几何误差最小化的算法应该给出MLE的理论边界。另一种最小化不同代价函数（如代数误差）的算法可以根据计算它离MLE给出的边界的距离来得到。

general estimation problem关注的是从到的函数f，其中是参数空间，是测量空间。假设点，向量,则有。x,y组成了点，其中i=1,....,n，对应着N维向量，其中N=2n，homograph的参数由向量P构成，homograph可能是8维向量或者9维向量，其值取决于。

通过均值为，方差为 的isotropic Gaussian distribution选择measurement vector X。参数向量P随着邻居point 的改变而改变，函数f(P)的值通过point X追踪中的surface SM。如图5.1中说明的那样，surface通过f的range给出。Surface的维度作为的submanifold，其值等价于d，d是自由度数量的关键参数，在计算single image的error中，d=8，这是因为由矩阵H确定的mapping具有独立缩放。

现在给定measurement vector X，利用ML算法估计出的接近于x的x（位于曲面SN上）。ML估计器是返回这个曲面上的最接近于X的点，用表示这个ML估计值。我们现在假设在X的邻域，该表面本质上是平面的，并且可以很好地近似于切面。在这个线性近似中，ML估计出的是从X垂直到切平面的foot。残差是从点X到估计值的距离。此外，从到X的距离是从最优估计值到真实值的距离，如图5.2所示。我们的任务是计算这些error的期望值。

计算ML残差的期望值现在已经被抽象为一个几何问题如下。一个N维高斯分布的总方差是协方差矩阵的轨迹，即在每个轴向方向上的方差之和。当然，这是由于正交坐标系的改变而没有改变的。一个N维高斯分布的total variance是covariance matrix的trace，每个轴方向的variance的和将得到total variance。对于independence variance 为 的isotropic Gaussian distribution，总方差为。现在，给定一个定义在上随机的总方差为，均值为，isotropic Gaussian distribution，我们希望通过计算出随机变量从到维度为d的超平面的期望距离。中的Gaussian random variable在d维切平面上的projection给出了estimation error的分布。在（N-d）-dimensional normal对切面的projection给出了残差的分布。

**Result 5.1** 定义在上的total variance为的Gaussian distribute投影到维度为s的subspace等价于具有total variance为的isotropic Gaussian distribution。

这一点的证明很简单，但被省略了。我们将此应用于s = d和s = N−d的两种情况下，得到以下结果。

**Result 5.2** N个测量值依赖于包含d个essential parameter的函数确定。假设测量值受到independence Gaussian noise的影响，每个测量变量的标准偏差为σ，则可以得到下面两个结论。

（1）RMS残差



（2）RMS estimation error



**2D homograph-一张图片中的error：**对于本章所考虑的二维投影估计问题，假设只有第二幅图像有误差，且d = 8和N = 2n，其中n是point match的个数。则可以得到如下残差和RMS estimation error：





在两幅图片中的error：在这种情况下，N = 4n和d = 2n + 8。假设tangent surface在真true measurement vector 附近呈线性关系，则可以得到如下残差和RMS estimation error：





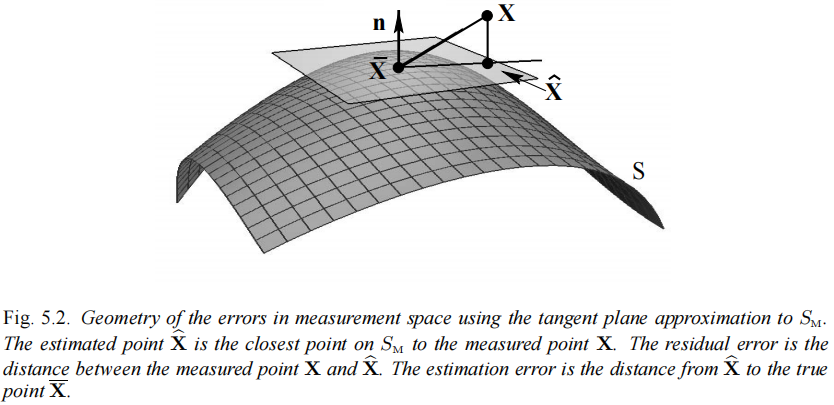
从这个图中可以得到的一个有趣的观察是，true value的asymptotic error是，而不是像一幅图像中的error那样是0。这个结果是意料之中的，因为，一个是每个point的位置都有两个measurement。对于一个point的两个measurement，点位置估计的方差减少了。相比之下，一张图片的error中，每个点都对应一个精确的测量值，这样，随着transformation H的估计精度越来越高，第二幅图像中的点的精确位置变得已知，asymptotic error渐近于0。

**Mahalanobis distance：**上述公式是在measurement space的error distribution是属于isotropic Gaussian distribution的情况下推导出来的，这意味着每个坐标的误差都是独立的。这个假设并不是关键。我们可以假设任何error的Gaussian distribution，并带有协方差矩阵Σ。结果5.2的公式仍然正确，被预期的马氏距离所取代，由于**Mahalanobis distance**此时标准偏差σ也消失了。

### 5.1.4 确定一个算法convergence的正确性

（5.3）和（5.4）给出了一种简单的确定estimate algrithm的正确收敛性的方法，而不需要确定问题的自由度数。如图5.2所示，measure space对应的model通过parameter vector P确定，最终形成surface 。如果在无噪声数据附近的surface接近平面，那么它可以近似于它的切平面，并且三个点，和形成一个right-angle triangle。在大多数估计问题中，这种平面性的假设将非常接近于通过typical noise magnitude矫正的scale set。在这种情况下，Pythagorean的等式可以写成

 (5.7)



在用合成数据（synthetic data）评估算法时，上面等式允许用一个简单的测试来查看算法是否已经收敛到最优值。如果估计值满足这个等式，则这表明算法已经找到了真正的全局最小值。请注意，无法用这个测试来确定自由度。下面还列出了以下一些属性：

* 此测试可以 run-by-run basis确定该算法是否已成功。因此，通过重复运行，它允许估计算法的成功率百分比。
* 该测试只能用于合成数据，或者至少是已知真实测量值的数据
* 该等式（5.7）由有效测量值组成的surface 确定，该等式代表的是局部平面。如果等式不满足特定运行的estimate algorithm，那么说明surface不是planar，或者（更有可能的是）因为算法没有找到最佳解。
* Test（5.7）是寻找全局解的测试，而不是寻找局部解。如果是局部最小，那么（5.7）的右侧很可能比左侧大得多。

## 5.2 estimate transformation的covariance

在前一节中我们使用极大似然估计值，以及计算expect average error。通过比较estimate error和residual error来评估算法性能的好坏。

然而，我们的主要关注点还是是transformation本身的计算精度有多准确。Estimation transformation的不确定性取决于许多因素，包括用于计算它的point的数量、给定point match的准确性，以及point的configuration。为了说明configuration的重要性，假设用于计算transformation的point接近degenerate configuration，那么计算出的transformation的计算精度可能不会很高。例如，如果变换是从一组靠近直线的点计算出来的，那么transformation还原出垂直于该直线的其他line可能就不会准确。因此，虽然estimate error和residual error被认为只依赖于point correspondence的数量及其精度，但计算transformation的精度也要依赖于特定的点。Transformation的uncertain很容易从covariance matrix中捕获到。H是一个有9个项的矩阵，它的协方差矩阵将是一个9\*9的矩阵，我们将会看到这个协方差矩阵是如何计算的。

### 5.2.1 covariance的正向传播

协方差矩阵在仿射变换下的计算十分简单。

**结论5.3：**设v是中的一个随机向量，该向量的均值为,协方差矩阵为，假设是affine mapping，其中。那么是一个随机变量，其均值为和协方差矩阵的随机变量。

请注意，它并不假设A是一个方阵。我们没有给出这个定理的证明，而是给出了一个例子。

**例子5.4：**设x和y是均值为0，标准差分别为1和2的独立随机变量。那么的平均值和标准差是多少？

均值为，接下来我们计算的方差。因为，,的方差为，最后得到标准差为5。

**例子5.5：**令。寻找的协方差矩阵，假设x和y的分布和以前相同。此时矩阵，则可以计算得到，进而得到的方差为25（标准差为5），其中和是负相关的，协方差。

**非线性propagation（传播）：**如果v是中的一个随机向量，是作用在v上的非线性函数，然后，我们可以计算出的均值和协方差的近似值，假设是分布的均值附近的近似仿射在。f的仿射近似表达式为，其中J为雅可比矩阵，J的维度为N\*M，进而我们可以得到下面结论。

**结论5.6：**令v是中的一个随机向量，该向量的均值为,协方差矩阵为，并且令在的领域范围内可微。然后到一阶近似，是一个均值为，协方差为的随机变量，其中J是f的雅可比矩阵，在处进行evaluate。

最终结果得到实际均值的近似值，的方差值取决于函数f与一个近似的线性函数有多接近，该线性函数在的commensurate区域，size与v的概率分布相匹配。

**例子5.7 ：**设是一个均值为和协方差矩阵为的高斯随机向量。让。然后可以根据公式计算的平均值和标准差的true value



其中：



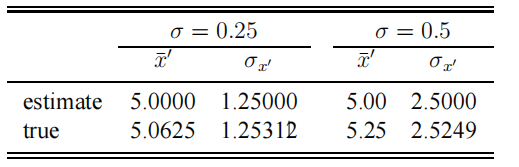
上式为高斯概率分布（A2.1-p565）。从而可以得到：



结果应用到5.6，并注意到J = [3−2]，我们可以得到估计值为



因此，只要σ足够小，这就很接近于的均值和方差的correct value。下表显示了两个不同的σ值下的均值和标准差的真实值和近似值。



作为参考，在σ = 0.25的情况下，只要（约占总分布的95%），值与其线性近似的差异不超过。

例子5.8：更一般地，假设x和y是独立的零均值高斯随机变量，可以计算函数的均值和方差如下：



上面计算的值接近于估计值均值=f，方差=，只要σx和σy足够小即可。

### 5.2.2 协方差的反向传播

本节中的材料和5.2.3节中的材料更加先进。第5.2.4节中的示例展示了这些部分的结果的直接应用。

考虑一个可微映射f从一个“参数空间”，从到measurement space ，定义在中的Gaussian probability distribution其具有的协方差矩阵用表示。设为映射f的图像。我们假设M<N，并且与参数空间具有相同的维度M。因此我们不用担心过度参数化问题。在中向量P表示上的点f (P)的参数化表达。在上找到距离点X Mahalanobis 距离最近的点，点X位于中的上。我们称这个映射为η： 。现在，假设f在表面上是可逆的，则可以定义为逆函数。

通过组合地图η： IRN→SM和f−1：SM→IRM，我们有了一个地图f−1◦η： IRN→IRM。 通过组合和，我们可以得到映射。这个映射给出一个测量向量X，以及用ML方法估计出的所对应的一系列参数P。原则上，我们可以在测量空间中传播probability distribution的协方差，通过协方差计算与ML estimation对应的parameter P的协方差矩阵。我们的目标是应用到结论5.3或者5.6中。

我们首先假定mapping f是从到的仿射映射。接下假定mapping 的映射也是一个仿射映射，对于将给出一个特定的形式，然后利用 result 5.3来计算估计estimated parameter的协方差。

因为f是affine，,其中表示上probability distribution的均值。因为我们假设surface 的维度为M，J的秩等于它的列维数。给定一个measurement vector X，ML算法通过最小化来估计。我们寻找来最小化后一个数量。然而：



此时，下式将在A5.2.1节中最小化





是协方差矩阵

**Result 5.9 backward transport的协方差——affine case：**公式表示的仿射映射，其中J的秩为M，设X是中的一个随机变量，均值为和协方差矩阵为。令表示measurement X到的映射。表示随机变量，均值为和协方差矩阵为

在f不是仿射的情况下，可以通过用仿射函数近似f来得到均值和协方差的近似，如下所示。

**Result5.10 协方差的backward transport——非线性case：**表示differentable mapping，J为雅克比矩阵，用于估计，假设J的秩为M，则f是表示的一对一neighborhood。设X是中的一个随机变量，均值为和协方差矩阵为。令表示measurement X到的映射。则在first-order，随机变量的均值为，协方差矩阵为。

### 5.2.3 over parameter

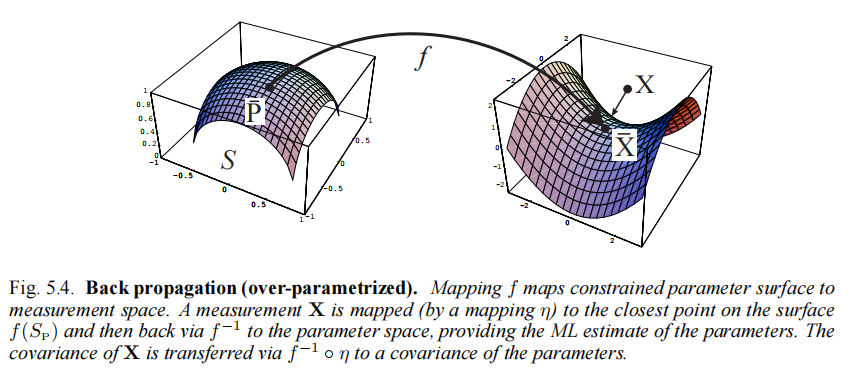
将结果5.9和结果5.10推广到冗余参数集的情况——over-parameter的情况。在这种情况下，从parameter space 到measurement space 的mapping不是局部一对一的。例如，计算2D homography，mapping 为f(P)，计算出P是9维向量，是homography matrix H的entries。因为homography只有8个自由度，mapping f不是一对一的mapping。对于任意常数k，矩阵kH代表相同的映射，因此image coordinate vector f (P)和f（kP）是相等的。

Mapping 中，雅克比矩阵J的秩为d，d<M。秩d被称为essential parameters。矩阵的维度为M，但是秩d<M。所以不能成立，因为等式右边的矩阵不可逆。

事实上，很明显，在没有任何进一步限制的情况下，estimate vector 的元素可以通过乘以任意常数k，没有约束地任意变化。因此元素具有无限方差，通常通过施加某种约束来限制estimated homography matrix H 或more generally parameter vector P。通常的约束是P = 1。 因此，parameter vectorP被限制在parameter space 或者 generally 中的一个表面上。Surface 表示中的unit sphere。Constraint 表示中的plane。在一般情况下，我们可以假设估计的向量P位于的某个submanifold上

**Result 5.11 covariance的反向传播——over-parameter case：**令为differentiable mapping，将parameter vector 映射到measurement vector。令为上经过点P的smooth manifold，在manifold 上的mapping是one-to-one映射的。Function f是局部可逆的，用表示，只在附近的surface 上可逆。令上的高斯分布的均值为，covariance matrix为，令为中point到中point的映射，中概率分布的的covariance matrix为的covariance matrix为

其中A表示m\*dmatrix，A的列向量在P处span 的切空间，如图5.4所示，



**证明：**result5.11的证明很简单。设d是essential parameter的个数。令为的U到上的的映射关系。Combine mapping在neighbourhoodU上时一对一映射。通过J计算f的partial derivative matrix，通过A计算g的partial derivative matrix，通过JA计算的partial derivative matrix。Result 5.10的应用，可以得到，上probability distribution的covariance matrix为，经过transported backward后得到上的covariance matrix为。应用result 5.6，在上transport forward得到covariance matrix。矩阵的pseudo-inverse用表示。只要A的列跨度不变，表达式（5.9）就不依赖于矩阵A的特定选择。如果A被AB取代，其中B是d\*d的可逆矩阵，则（5.9）的值不会改变，这样，任意matrix A的列在的tangent space上spam。

注意，该证明给出了g的雅可比矩阵。有更简单的方法找到A。令维数为M。秩为d，d<M。这是因为estimate parameter set的variance为0，注意是不可逆的，是d\*d的矩阵，秩为d，可逆。

当constraint surface与雅可比矩阵的null-space局部正交时，就发生了一个重要的情况。令表示矩阵X的left null-space，被称为所有向量x的space，pseudo-inverse用下式表示：



Result5.12令表示到的映射，令J表示f的雅可比矩阵。令上的高斯分布的均值为，covariance matrix。为measurement X到MLE parameter vector P的映射。包含上分布的covariance matrix，



注意，P被限制在一个surface与J的null-space的局部正交上。例如，如果P是homogeneous parameter vector，则usual constraint 可以满足restriction。在这种情况下，约束面是单位球面，并且在任何一点上的切平面都垂直于参数向量。另一方面，由于P是homogeneous vector，所以function f(P)不会随着scale的改变而改变，并且J在radial 方向上是null-vector，因此，它垂直于约束面。

在其他情况下，为了计算参数的协方差矩阵，我们对参数集设置了一些限制。此外，由于pseudo-inverse操作得到它自己的逆，我们可以从它的pseudo-inverse中检索到原始矩阵后,就可以计算与任何其他子空间对应的协方差矩阵:



其中，A的列span了parameter space的受约束的子空间。

### 5.2.4 应用和例子

**Error in one image：**我们开始考虑将这个理论应用到estimate 2D homography H的covariance问题。首先，我们看看误差仅限于第二幅图像的情况。H是一个3\*3的matrix，写成9维parameter vector，用h表示这个9维parameter vector。Estimate 的covariance是一个9\*9的symmetric matrix。给定一组matched point。Point 代表fixed true value，point代表受variance或者covariance为的高斯噪声影响的random variable。Function代表用9-vector h表示的matrix H到2n-vector的映射，2n-vector由组成。的坐标由中的composite vector组成。正如我们看到的，随着h的变化，point f(h)在In中追踪出一个8维的表面。曲面上的表示与第一张图片中point 对应的point 的集合。给定一个measurement vector ，选择位于上的点，该点距离有最小的mahalanobis 距离。Pre-image，具有约束，h表示estimate homography matrix。从具有probability distribution中估计出。Covariance matrix 可以从result 5.12中得到，这个covariance matrix 对应的constraint 为。

计算estimate transformation的covariance matrix的过程如下：

1. 从给定的数据中计算出estimate transformation
2. 从中计算出雅可比矩阵
3. 计算出estimate h的covariance matrix 

我们将更详细地研究这种方法的最后两个步骤

**Derivative matrix的计算：**对一个雅可比矩阵为，这个矩阵可以被分解成block，可以用表示。已知：

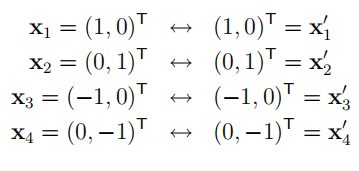
 （5.11）

其中表示向量

将这些矩阵堆叠在所有点上，得到derivative matrix。当image measurement 是一个random vector时，中的每个都是2\*2的covariance matrix，中的第i个measure point 为，则可以计算出

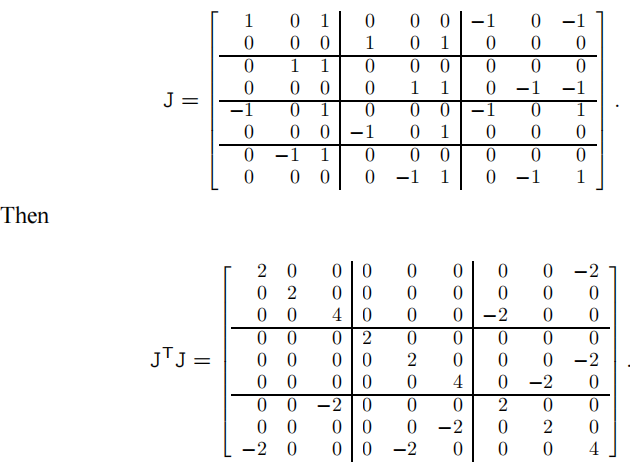
 （5.12）

Example 5.13：我们考虑一个只包含4个point correspondence的例子如下：

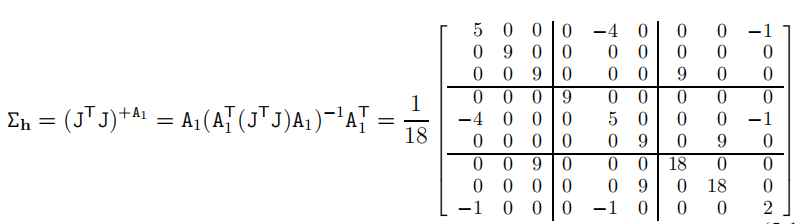


假设point 是已知的，且每个point 都有一个像素的standard deviation，则covariance matrix是唯一的。

显然，计算出的homography将是identity map。为了简单起见，我们对它进行归一化，使它成为identity matrix，这里的归一化为而不是通常的归一化H = 1。在这种情况下，所有的在（5.11）中都等于1。矩阵J很容易从（5.11）计算到：

 （5.13）

为了取这个矩阵的pseudo-inverse，我们利用（5.9）结论——A是一个矩阵，其列跨越constraint surface的tangent plane。由于H的约束条件为，因此约束面垂直于homography H对应的vectorh。h的Householder matrix A具有的特性，因此的前8列根据需要垂直于h。这允许在不使用SVD的情况下精确地计算pseudo-inverse。应用（5.9）将伪逆计算为



对角线给出了H项的个体方差。

这个计算出的协方差用于评估点转移的准确性。

### 5.2.5 两张图片中的error

在两幅图像都有误差的情况下，计算transformation的协方差要复杂一些。如第4.2.7节（p101）所示，可以定义一组2n+8参数，其中8个参数描述变换矩阵，2n个参数表示第一图像中的point estimate。更简单的方法是对H进行over-parameter。雅可比矩阵自然地分为两部分，，其中A和B分别是camera parameter和point 的导数。计算如下：



这个矩阵的pseudo-inverse是parameter set的协方差，这个pseudo-inverse的左上角块是H entries的协方差。第A6.4.1节（p608）对此进行了详细的讨论，其中展示了如何利用雅可比矩阵的块结构来简化计算。

### 5.2.6 在point transfer中使用covariance matrix

一旦有了协方差，就可以计算point transfer。第一张图像中的new point x，并不是用在计算H上。第二幅图像中correspond point是。然而，由于H的估计存在不确定性，point 的correct location也会有不确定性。我们可以用H的协方差矩阵来计算这个不确定性。

point 的协方差矩阵由下面公式给出

 (5.16)

雅可比矩阵为。协方差矩阵用H的协方差矩阵表示。从（5.9）中，这个协方差矩阵Σh依赖于estimating H时所用的particular constraint。因此，Σx似乎也依赖于用于约束h的特定方法。

**Example 5.14 ：**继续例子5.13，2D homography H由单位矩阵求出，协方差矩阵如（5.14）所示。Point（x，y）是上的点。在这种情况下，协方差矩阵等价于下面公式：



注意，和是x和y的even function，而是一个old function。Point 关于x轴y轴对称用于计算H。如果x和y交换，和也会改变。

可以看出，variance随x的四次幂而变化，因此standard deviation随平方而变化。这表明，transform point 的extrapolating远远超出的用于计算H point set。的RMS为，这个等式等价于，其中r是到原点的径向距离。请注意一个有趣的事实，即RMS error只依赖于径向距离。

本例计算了transfer point在最小情况下的covariance。对于四个以上的correspondence，情况并没有本质上的不同。超出 Extrapolation的point setj计算出的H是不可靠的。事实上，可以表明，如果H是由一个单位圆均匀间隔的n个点计算的，则RMS error为，所以误差呈现出相同的二次增长。

## 5.3 Monte Carlo estimation的协方差

前几节中讨论的协方差估计方法依赖于线性假设。换句话说，假设surface f (h)在估计点附近是局部平坦的，至少在与噪声分布的近似范围相对应的区域上。它还假设了变换H的估计方法是最大似然估计。如果surface不是完全平坦的，那么协方差的估计可能是不正确的。此外，particular estimation method可能不如最大似然估计，从而在估计的变换H的值中引入额外的不确定性。

一种更一般的（虽然昂贵的）估计协方差的方法是exhaustive simulation。假设噪声是从给定的噪声分布中提取的，那么从一组perfectly match correspond point开始。然后将噪声添加到点上，并使用所选择的估计程序计算相应的变换。Transformation H的协方差从多次试验统计的符合噪声分布的点求出。

Analytical和Monte Carlo method估计H的协方差都可以应用于在不知道H真实值的情况下从真实数据中估计协方差。从给定的数据，估计H和相应的和。协方差的计算的前提是estimated value非常接近true value。假设所得到的协方差矩阵为真实变换的协方差。这种识别是基于这样一个假设，即数据点的真实值足够接近于估计值，因此协方差矩阵基本不受影响。

## 5.4 closure

### 5.4.1 文献

通过只使用一阶泰勒展开式，并假设有高斯误差分布，贯穿本章的推导已经大大简化。类似的想法（ML，协方差…）可以为使用费雪信息矩阵的其他分布。相关阅读可在卡那尼、出版社等人和其他统计教科书中找到。

Crinisi等人[犯罪-99b]给出了许多H中计算协方差的例子，因为用于确定同系性的对应在数量和位置上不同。

### 5.4.2 笔记和练习

(1)考虑利用正交回归计算平面上一组二维点的最佳线拟合的问题。假设N个点在每个坐标上用σ的独立标准差来测量。每个点与拟合线的预期均方根距离是多少？

答案：

# Part 1 :相机几何和单目几何——camera geometry and single view geometry

第6章描述了 3D scene space到 2D image plane上的投影。camera mapping由一个矩阵表示，mapping points是一个3×4矩阵P，它将 3-space中一个世界点的齐次坐标映射到image plane上的齐次坐标。这个矩阵一般有11个自由度，而相机的性质，如它的中心和焦距，可以从矩阵中提取出来。特别是照相机的内部参数，如焦距和长宽比，被包装在一个3×3的矩阵K中，这是通过P的一个简单分解得到的。有两种特别重要的相机矩阵：finite cameras和cameras with their centre at infinity，如表示parallel projection的affine camera。

第7章描述了 通过给定的一组相应的世界和图像点的坐标估计camera matrix P。本章还描述了如何有效地将相机的约束纳入估计，以及一种校正 radial lens distortion的方法。

第8章有三个主要主题。首先，它涵盖了照相机对finite points以外的几何物体的作用。这些点包括lines, conics, quadrics and points at infinity。在points/lines at infinity是消失的点/线。第二个主题是camera calibration，在不计算整个矩阵P的情况下对相机的internal parameters K矩阵计算。特别描述了internal parameters与 absolute conic的关系，以及摄像机从vanishing points和vanishing lines的校准。最后一个主题是calibrating conic。这是一个简单的geometric device，其可以可视化 camera calibration。

# Camera model

照相机是3D world（对象空间）和2D image之间的映射。这本书中最感兴趣的是central projection。本章开发了许多 camera models，它们是具有camera mapping的特定属性的矩阵。可以看到，所有 cameras modelling central projection都是general projective camera的专门化。这个最general camera model的解构可以使用projective geometry工具来检测。可以看到，相机的 geometric entities，如 projection centre和 image plane，可以很简单地从其 matrix representation中计算出来。general projective camera的Specializations继承了它的属性，例如，它们的几何形状使用相同的代数表达式进行计算。specialized models主要分为两类：一类是finite centre的相机，另一种是 centre “at infinity”的相机。在infinity camera中，affine camera特别重要，因为它是parallel projection的自然推广。本章主要涉 及各点的投影。照相机对其他 geometric entities的作用，如线，将被推迟到第8章。

## 6.1 finite camera

在本节中，我们从最专业和最简单的相机模型开始，即基本的针孔相机，然后通过一系列的渐变逐步推广这个模型。我们开发的模型主要是为CCD等传感器设计的，但也适用于其他相机，如x射线图像、扫描的摄影底片、放大底片的扫描照片等。

The basic pinhole model.（针孔模型）：我们考虑空间中点在平面上的central projection。设central projection是欧几里得坐标系的原点，并令Z = f，这样的平面被称为image plane或focal plane。在针孔照相机模型下，space中的坐标用表示，该点映射到 image plane上，点X到projection centre的线与 image plane相交。通过相似三角形，point 映射到image plane上为。这个变换描述了从world coordinates到image coordinates的central projection mapping。这个mapping是将Euclidean 3-space映射到 Euclidean 2-space。

Centre projection 也叫做 camera centre，这也是optical centre。从camera centre垂直于image plane的直线叫做相机的principal axis或者principal ray，principal axis 与image plane的点叫做principal point。穿过camera centre且平行于image plane的plane叫做principal plane。

* **使用齐次坐标的中心投影：**

如果世界点和图像点用齐次向量表示，那么central projection就是homogeneous coordinates之间的线性映射。写成如下形式

 （6.2）

这个表达式中的矩阵可以写成，其中是一个对角矩阵，[I | 0]表示一个矩阵被分成一个3×3块（单位矩阵）加上一个列向量，这里是零向量。X由4维齐次向量表示world point，x由3维齐次向量表示image point，P表示3×4齐次camera projection matrix。然后（6.2）被紧凑地写为



它将中心投影的针孔模型的camera matrix定义为:



* **Principle point offset：**









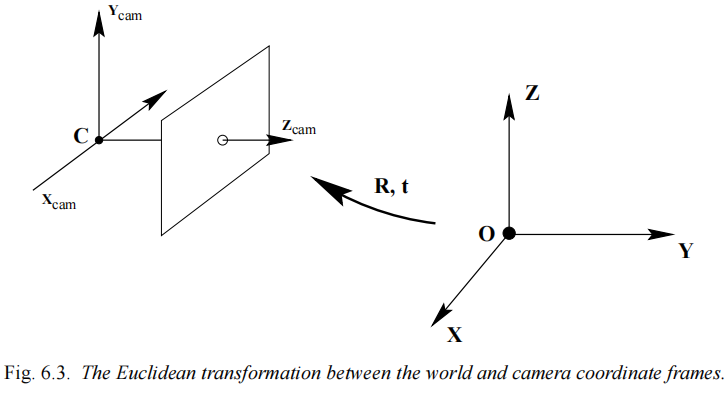
K——内参（camera calibration matrix），

——用表示，表示相机坐标系（camera coordinate frame）

——principal point在相机坐标系下的坐标

——uv坐标，image coordinate下的uv坐标

* **Camera rotation和translation：**



Euclidean coordinate=world coordinate frame

：world coordinate frame下的point A的非齐次坐标

：camera coordinate frame下point A的非齐次坐标



：表示world coordinate frame下的camera centre坐标

 ： rotation matrix ，表示camera coordinate frame的旋转

：world coordinate frame中坐标，



：P有9个自由度，3个是K，3个是R，3个是；R和组合起来就是外参（external parameter或者 exterior orientation）



：P 所有参数，其中

* **CCD camera：**

：

：x方向上，image coordinate中每单位距离上pixel的个数

：y方向上，image coordinate中每单位距离上pixel的个数

f：image coordinate frame中的距离下的focal length

：在像素尺度上，x方向上的focal length

：在像素尺度上，y方向上的focal length

：在像素尺度上，principal point的坐标





* **Finite projective camera：**

：s为skew parameter，在大多数的normal camera中的skew parameter都为0，在特殊情况下，s不为0。

：P有11个自由度。

：K是P中的上三角（upper-triangular），R是rotation matrix



* **General projective camera：**

K为任意齐次3\*4矩阵，秩为3，11个自由度，（秩为3的原因为：如果秩小于3，那么矩阵映射的范围将是一条线或一个点，而不是整个平面；换句话说，它不是一个二维图像。）

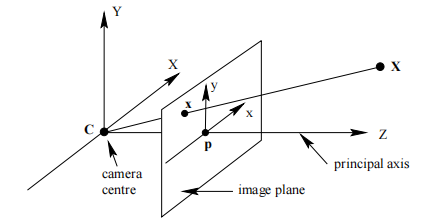
## 6.2 projective camera

：

：image point

：image point

### 6.2.1 Camera anatomy





M：3\*3matrix，如果M为non-singular则对应finite camera；如果M为singular，则对应infinite camera

* **Camera centre：**

：P参数空间；C表示camera centre，用四维齐次向量表示；

：

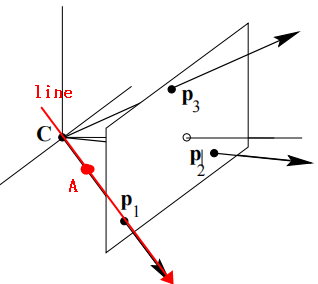
: 相机参数空间

:

：

：M是P的submatrix 是3\*3的singular矩阵

：point A在line上，这个line经过camera centre C。因为。该line上的所有点都映射到同一个图像点PA，这意味着该line必须是通过camera center的射线。



：

* **Column vector**

: 相机所有参数，内外参

：P的每一列，P1,P2,P3是world coordinate中 x轴，y轴，z轴落到image中的成为vanishing point；P4表示world origin

：世界坐标系中，x轴方向

：x轴在image中的坐标

* **Row vector**

: 每一行表示一个particular world plane

* **Principal plane**

Principal plane穿过camera centre 且平行于image plane。

：line延伸到无穷远落在infinity plane上变成一个点X，点X组成principal plane。

: 是P的每一行。

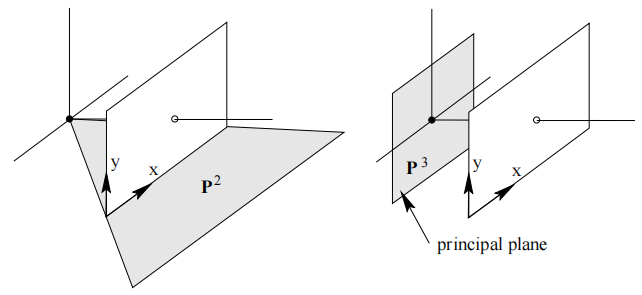
：当时，point X落到principal plane上

：表示camera的principal plane的向量

：C表示camera centre

：

：表示C在principal plane上



* **Axis plane**

：point

：plane

：point X和plane 满足

：表示point X，落在image的y-axis

：当PC=0时，C落在plane 上

因此，平面由图像中的camera centre和图像中的线x = 0定义。类似地，平面由camera centre和线y = 0定义。与principal plane 不同，axis plane 和依赖于图像的x轴和y轴，即依赖于图像坐标系的选择。因此，它们与natural camera geometry紧密程度不如principal plane 。特别地，axis plane 和的交点线是连接camera centre和image origin的线，即image origin的反投影。这条线一般不会与相机的principal axis重合。

camera centre C位于所有三个平面上，并且由于这些平面是不同的（因为P矩阵的秩为3），所以它必须位于它们的交点上。在代数上，camera centre位于所有三个平面上的条件是PC = 0。

* **Principal point**

：是一个plane

：表示的normal vector

：表示infinity plane上的一个point

：camera上的principal plane为 ，位于infinity plane上的point为，用符号

：计算得到的是camera的principal point

：M：3\*3matrix，如果M为non-singular则对应finite camera；如果M为singular，则对应infinite camera

：计算得到的是camera的principal point，其中是M的第三行。

* **Principal axis vector**

: point X映射到image中成为x

: 相机参数

：是principal axis上的point

：camera coordinate frame中3D point X到image x的映射

：是在principal axis方向上的一个向量，指向摄像机的前面。

：

：

### 6.2.2 projective camera 上的point

* **Forward projection：**

: image中的point

: space point

:

：point D=vanishing point

：

：M是P的3\*3submatrix

* **从point到ray的back-projection：**

给定图像中的一个点x，我们接下来确定空间中映射到这个点的点的集合。这一套将构成空间中穿过相机中心的射线。光线的形式可以通过几种方式来指定，这取决于人们希望如何在3个空间中表示一条线。原告的代表被推迟到第8.1.2节（第196页）。在这里，这条线被表示为两点的连接点。

: camera centre PC=0

: 相机参数，

：的pseudo-inverse



：和的关系

：point 在ray上

：证明是的project，且在ray上

：ray是由和组成的line

：camera centre的坐标

：image point back-project到ray上与infinity plane相交于point D，D是位于ray上的第二个点

：再次把这条线写成射线上两点的连接处

### 6.2.3 point 的depth

计算point距离principal plane的距离

: 相机参数，camera matrix

：在3-space中的point

：将image point 映射到X

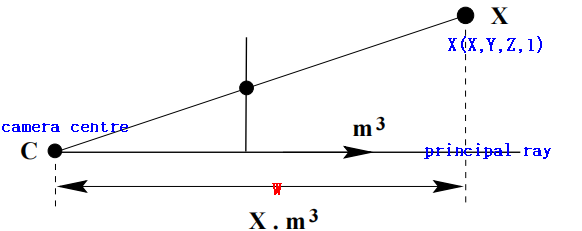
：camera centre

：因为PC=0，所以有

：为principal ray direction

： w表示为从camera centre到point X的射线与principal ray direction 的点积 ，w为点X从相机中心C在principal ray方向上的深度。

: 当camera matrix 被归一化后，可以得到，为unit vector pointing



Result 6.1

:3D point

：finite camera 的camera matrix

：一个假设

：基于上面假设计算出X在principal plane前面的深度

这个公式是确定一个点X是否在相机前面的有效方法。一个验证方法是如果point X或camera matrix P乘以一个常数因子k，看的值是否不变。因此，独立于X和P的特殊齐次表示。

### 6.2.4 camera matrix 的分解（decomposition）

设P是一个表示一般射影相机的相机矩阵。我们希望从P中找到camera centre、camera orientation和camera internal parameter。

**找到 camera centre：**

:

: camera centre

：通过P求出camera centre C

* **找到camera orientation 和internal parameter：**

: finite camera

：将M分解成一个上三角矩阵K和正交矩阵R的乘积

：通过分解M得到R，camera orientation（相机方向）

：通过分解M得到K，calibration matrix（internal parameter）

: x-coordinate direction的scale factor

：y-coordinate direction的scale factor

：skew（偏移量）

：principal point的坐标

：aspect ratio

* **Example 6.2**

已知camera matrix



根据得到C和M

：camera centre

: 由P得到M，分解M得到内参K和camera orientation R

* **当K中的s不为0：**

如第6.1节所示，一个真正的CCD相机只有四个内部相机参数，因为通常是s=0。如果s不为0，意味着CCD array中pixel element倾斜，从而使x轴和y轴不垂直。诚然，这是不太可能发生的。

: 假设origin（finite） camera用matrix P表示

: homography matrix

: 表示一张图片；因为H是non-singular的，所以HP的left 3\*3 submatrix也是non-singular的，则即使s不等于0也可以分解得到KR。但是此时计算出的K不再是calibration，R不再是original camera

* **什么情况下需要decomposition：**

:当时，K,R,C的值都已知了，这个时候就不需要decomposition。

Camera： camera是从world和image的对应点中计算出，并间接地通过计算多重视图关系（如基本矩阵或三焦张量），然后从这个关系计算投影矩阵。此时就要用到decomposition。

* **Coordinate orientation的注意事项**

世界和图像坐标之间的关系仍然用3×4相机矩阵来表示。在和为正的情况下，仍然可以根据对camera matrix P进行分解得到K。不同的是，R现在表示camera orientation为z轴负方向。此外，由给出的点的深度对于相机前的点将是负的，而不是正的。如果记住了这一点，那么就允许在图像中使用左手坐标。

### 6.2.5 euclidean space和projective space

目前一直假设world coordinate和image coordinate是Euclidean。借鉴projective geometry的思路（如π∞上点对应的方向），homography coordinate的符号线性的表示central projection。在本书的后续章节中，我们将进一步使用projective coordinate。因为假设世界坐标系是projective 的，

：transformation between the camera and world coordinate frame

：map from projective 3-space IP3 to the image

事实上，在最一般的情况下，projective camera是从到的映射，它涵盖了3-space的projective transformation、从3-space到image的投影、图像的projective transformation，这三个的合成效果。

：P与homography的关系

然而要记住相机是欧几里得设备，不能因为我们有一个projective model camera就避欧几里得几何的概念

欧几里得法和仿射法：

虽然（finite）3×4矩阵总是可以像第6.2.4节那样分解以获得rotation matrix R、calibration matrix K，但只有当图像和空间坐标处于适当的框架内时，所获得的参数的euclidean interpretation才有意义。在分解的情况下，对于image和3-space都需要一个euclidean frame。另一方面，即使both frame都是projective frame，也可以将P的null-vector作为camera centre——interpretation只需要collinearity，这是一个projective notion。将作为principal plane前提是需要image和3-space的affine frame。最后，将作为principal ray的前提是需要一个affine image frame，另一方面euclidean world frame在计算orthogonality方面也有重要意义。

## 6.3 infinity camera

：infinity camera的分类

### 6.3.1 affine camera

Affine camera的camera matrix P中最后一行为（0,0,0,1）

: 表示finite projective camera的camera matrix

: 旋转矩阵R的第i行。

：camera的坐标

：orientation matrix

：internal parameter matrix/ calibration matrix

：camera的principal ray

：principal ray上，world origin到camera centre的距离

Camera centre沿着principal ray移动 移动时间t，此时更新为

：

：，

: 经过时间t后，world origin到camera centre的距离

：camera focal的缩放因子

：图像大小保持固定



：趋近于infinity

### 6.3.2 affine camera中的error

：经过world origin且垂直于principal axis 的平面，不会受zooming和motion的改变

：表明X位于经过world origin且垂直于principal axis 的平面上，下的图像都是一样的

: 表示point X不在上述平面上，下的图像是不同的。

：表示通过得到的point

：表示通过得到的point

：calibration matrix

：进一步推导得到另一种表达方式

：进一步推导得到另一种表达方式

：第一个元素除以第二个元素得到

：第一个元素除以第二个元素得到



: 说明通过true perspective image获取的位置和通过affine camera approximation 获取的位置，两个位置很接近

一般来说，使用较长焦距的透镜获得的图像往往满足这些条件，因为视场和景深都小于使用相同CCD阵列的短焦距透镜获得的图像。对于在不同深度有许多点的场景，仿射相机不是一个很好的近似。例如，如果场景包含近距离的前景和背景对象，则不应该使用仿射相机模型。

### 6.3.3 decomposition of

：camera matrix，R是由rotation matrix 前两行决定，0是0向量，t是,是上三角

：用替代，其中

: 对或˜进行适当的替换，我们可以得到两种affine camera matrix

：第一种affine camera matrix

：第二种affine camera matrix，第一个矩阵是相机内参，第二个矩阵是相机外参。

和finite camera的本质区别：

1. parallel projection matrix代替了finite camera的典型投影矩阵[I | 0]
2. Calibration matrix取代了一个finite camera的K
3. 没有像finite camera那样定义principal point

### 6.3.4 affine camera的层次

与6.1节中finite projection camera分类的发展类似，我们可以从 parallel projection的基本操作开始，建立一个表示更一般的parallel projection的情况。

：沿z轴的投影，通过P将point 映射到image上

：对于一般的 orthographic projection mapping，3D Euclidean coordinate change form

：orthographic projection matrix ，有5个自由度，转矩阵R三个参数，加上两个偏移参数t1和t2，特征是矩阵最后一行为零，前两行正交，t3 = 1。

：scale orthographic projection，有六个自由度；特征是一个矩阵M的最后一行为0，前两行orthogonal and of equal norm。

：weak perspective projection，可以想象成一个infinity camera在两个轴向图像方向上的比例因子不相等。它有七个自由度。最后一个矩阵M行为零，前两行正交

：affine camera，它有八个自由度，被看作是finite projective camera的parallel projection version

：full generality affine camera，它有8个自由度，对应于8个non-zero和non-unit matrix element。我们用表示左上角的2×3子矩阵。对affine camera的唯一限制是的秩为2。这源于对P的秩是3的要求。

：描述了affine matrix将3-space映射到image

：非齐次坐标系下，affine matrix将3-space映射到image

：

: image里的world origin

### 6.3.5 affine camera的更多属性

:公式表达空间中无穷远的平面被映射到图像中无穷远的点。假设照相机的principal plane是在无穷远处的平面，因为optical centre位于principal plane上，所以optical centre也位于无穷远的平面上

1. 当principal plane位于无穷远的平面上时得到的projective camera matrix为affine camera matrix
2. Parallel world line会被投影为parallel image line。这是因为Parallel world line在无穷远处的平面上相交，这个交点被映射到图像中无穷远处的一个点。因此，image line是平行的。
3. 满足的向量d平行于projection方向，为camera centre

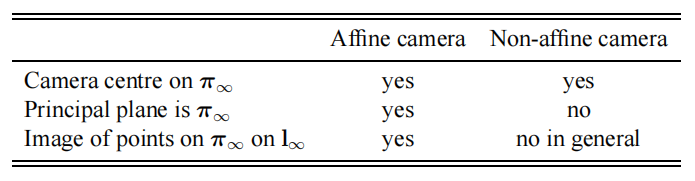
任何由仿射变换（空间或图像）的合成效果组成的相机都将具有仿射形式。例如，对透视投影由两个这样的映射组成：第一个是通过质心在平面π上的平行投影，并平行于映像平面。平行投影的方向是将质心连接到相机中心的光线。这个平行投影之后是π和图像之间的仿射变换（实际上是相似性）。因此，准透视相机是一个仿射相机。

### 6.3.6 infinity general camera

Principal plane在无穷远平面时此时是affine camera，它的camera centre也在无穷远平面上。然而，也会有camera centre在无穷远平面上，但是principal plane不在无穷远平面的情况。

如果P = [M | p4]，且M为一个奇异矩阵，则camera centre在无穷远平面上。如果M是奇异的，但M的最后一行不是零，那么这个相机就不是仿射的，但也不是一个有限的射影相机。然而，这样的照相机是一个相当奇怪的物体，在这本书中不会被详细处理。在这两种情况下，camera centre都是投影的方向。此外，在仿射相机的情况下，所有non-infinite point都在相机的前面。对于non-affine camera space，由principal plane划分为两组点。

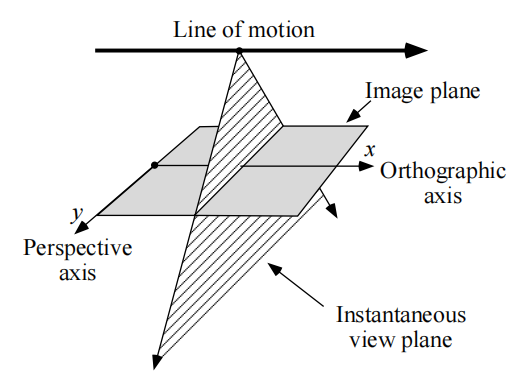
在无限远处的一般相机可以由仿射相机产生的图像的透视图像产生。这个成像过程是左乘一般的3×3矩阵，所得到的3×4矩阵在无穷远时仍然是一个照相机，但它没有仿射形式，因为世界上的平行线通常会在图像中以收敛线的形式出现。



## 6.4 其他camera model

### 6.4.1 pushbroom camera推杆式摄像机

Linear Pushbroom (LP) camera是卫星中常见的一种传感器的抽象，例如SPOT传感器，在这样的照相机中，线性传感器阵列用于一次捕获单行图像。当传感器移动时，传感器平面扫出一个空间区域（因此命名为推扫帚），一次捕捉图像一条线。该图像的第二个维度是由该传感器的运动来提供的。



### 6.4.2 line camera

一个平面的中心投影映射到1D line上。从平面的齐次表示到直线的齐次表示的线性映射。这个照相机有5个自由度。







## 6.5 closure

本章介绍了照相机的模型，它们的分类和分解。

设为projective image，为的图像（图像的图像）。用表示composite image。的apparent camera centre与相同。推测这解释了为什么肖像的眼睛“follow you round the room”。

**Affine camera：**

Affine camera是均匀坐标上最一般的线性映射，将parallel world lines映射到parallel image lines。考虑上的投影，当P具有仿射性质时，上的点会被映射到图像中无穷远处的点。

**The rational polynomial camera（有理多项式相机）：**

有理多项式相机是一种通用的相机模型，image coordinate定义如下：



finite projective camera P可以通过右乘4×4的homography H转换为正交相机（6.22）



# 计算camera matrix P

：通过3D point X和image point x的correspondence 计算出camera matrix P。这个计算过程类似2D projective transformation H的计算过程，只不过H是3\*3的矩阵，P是3\*4的矩阵。

## 7.1 basic equation

（7.1）：是4维向量，

（7.2）：只用左边矩阵的前两行

：p是camera matrix P的每一行，A是假设有n个point correspondence，每个左边矩阵有12个元素，堆叠起来一共有2\*n\*12个元素，写成2n\*12的矩阵A。

因为camera matrix P有12个元素，自由度为11，每对point correspondence能得到两个等式，则point correspondence的数量n=5.5就能得到11个等式，从而求出camera matrix P的精确解，四舍五入就是需求6对point correspondence。

：当n>=6时，会出现algebraic error，只能通过最小化来求P，最小化的约束条件为

：表示位于平面上，可以通过line和point correspondence来计算P

：表示image中的l被反投影为平面

## 7.2 geometric error

：accuracy world point

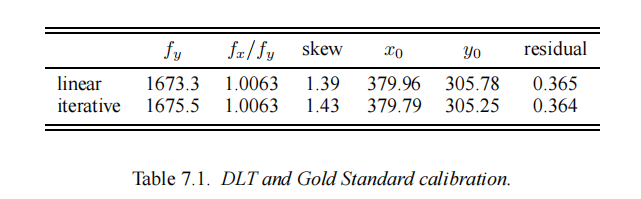
：measured point



：image中的geometric error

：P的maximum likelihood estimate

在2D-homography中，通过不断迭代求最小geometric error（例如levenberg-marquardt），在迭代求中可能将DLT解或者minimal solution作为迭代的起始点。

：DLT算法和gold standard算法均为用棋盘格标定相机计算内参时使用的算法

：3D geometric error

：同时考虑world error和image error，使用mahalanbis distance

**Gold standard 算法流程——当world point的坐标已知非常精确**

6对world-image correspondence point就可以确定projection matrix P的maximum likelihood estimate

* 第一步：求出线性解——
* 第二步：最小化geometric error——
* 第三步：denormalization——

**Gold standard算法流程——估计affine camera matrix** 

4对correspondence确定仿射相机的projection matrix 的maximum likelihood estimate，

算法步骤：

* 第一步：归一化，使用similarity transformation T对image point进行归一化，使用第二个similarity transformation U对space point进行归一化。Image point归一化公式为，space point的归一化公式为
* 第二步：根据n对correspondence堆叠得到矩阵等式



最后有2n\*8个等式，

* 第三步：求的伪逆得到P

其中

* 第四步：denomination，从中获得origin coordinate的camera matrix



### 7.2.1 geometric interpretation of algebraic error

：在DLT算法中归一化后的point

：在DLT算法中归一化后的point

：DLT算法中需要最小化的值



：表示point 在principal ray上到camera的距离



：与成比例

：f为focal length； 和一一对应，

：最小化algebraic error等价于最小化

：是的近似

### 7.2.2 estimation of an affine camera

Affine camera的projection matrix的最后一行为（0，0，0，1），仿射相机中的algebraic error和geometric image error是相等的

（7.5）：表示公式（7.2）在single correspondence 下的情况

：表示algebraic error和squared geometric error是相等的

## 7.3 restricted camera estimation

：R是3\*3的旋转矩阵，

（7.6）：P的内参部分，为了计算出P一般会假设（1）s=0（2）（3）principal point是已知的（4）内参K是已知的

* **最小化geometric error：**为了最小化几何误差，假设了如下约束：（1）s=0（2），使用iterative minimization（levenberg-marquardt）将参数集的几何误差最小化。
* **最小化 algebraic error：**用q表示P的9个参数，g表示map。所以有g=P(q)，最小化algebraic error等价于最小化
* **Reduced measurement matrix：**

：因为A是一个2n\*12的矩阵，当n很大时，A的行会很大，所以用12\*12的矩阵替代，叫做reduce measurement matrix

：对A进行SVD操作

：定义

：

:另一个获得的方法

：将映射到

：correspondence-》minimize algebraic distance->得到P

：需要被最小化，从而求出parameter q

：K是内参，P满足（因为这些条目与旋转矩阵R的最后一行相同）

* **Initialization：**

计算camera parameter需要使用迭代最小化error，迭代需要初始化

1. 用linear algorithm（DLT）初始化camera matrix
2. 设置s = 0，αx = αy=使用DLT获得的值的平均值
3. Decomposition initial camera matrix得到parameter

理想情况下，fix parameter的假设值将接近于由DLT得到的值。然而，在实践中情况并非总是如此，将这些参数改变到它们的期望值，会导致一个不正确的initial camera matrix，这可能会导致较大的残差，并难以收敛。

：soft constants——在s = 0，αx = αy=使用DLT获得的值的平均值的约束条件之外加入以此来解决initial camera matrix找不准确的问题，w表示权重

：cost function，首先从使用DLT估计的参数值开始迭代。权重从低值开始，并在估计过程的每次迭代中都会增加。因此，s的value和aspect ratio比被温和地绘制到它们期望的值。最后，它们可以被固定在它们期望的值上进行最终估计。

* **Exterior orientation（外参）：**

假设相机的所有internal parameters都是已知的，那么所有有待确定的就是相机的position and orientation (or pose)。这就是“exterior orientation”问题，这在calibrated systems的分析中很重要。为了计算exterior orientation，对在世界坐标系中已知精确位置的构型进行成像。然后再寻找照相机的姿势。这种情况出现在机器人系统的手眼校准中，其中需要相机的位置，也出现在基于模型的识别中，其中需要对齐物体相对于相机的位置。有六个参数必须确定，三个是orientation，三个是 position。由于每个world-image的correspondence产生两个约束，因此期望三个点就足够了。事实确实如此，所得到的非线性方程一般有四个解。

* **Experimental evaluation：**

最小化algebraic error和geometric error得到的parameter都有9个元素。然而，algebraic error最小化要快得多，因为它只最小化了12个误差，而不是在geometric error最小化中的2n = 396。

## 7.4 radial distortion

：是相机坐标系下的三维坐标，是图片坐标系下的坐标

：表示actual image position ，表示ideal image position，表示radial distance，表示radial distortion的中心，表示distortion factor



：是测量出来的点，是实际上真正正确的坐标，是radial distortion的中心。又与线性projection camera坐标系下的三维坐标直接相关。

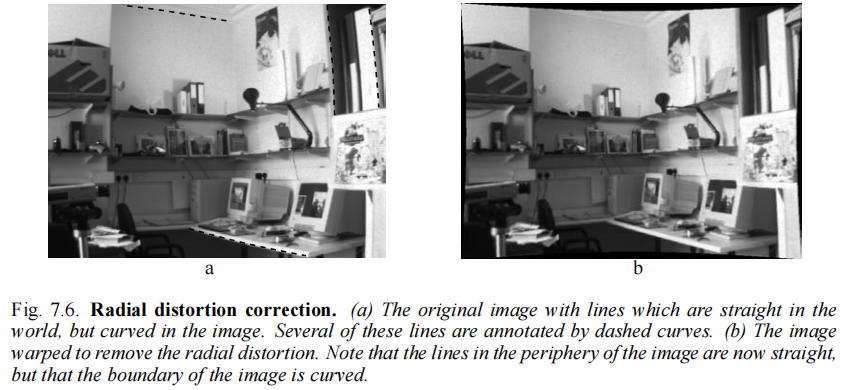


：的泰勒展开公式，

：对的矫正也是属于内参标定的一部分，此校正和camera calibration matrix都需要指定从图像点到相机坐标系中的映射。

去畸变函数可以作为成像过程的一部分，在迭代最小化geometric error的过程中，一起计算ki和P。更简单的方法是在场景中放直线，构建cost function来计算，它的优点是，由于场景提供了校准实体，因此不需要特殊的校准模式。

径向畸变矫正例子（1）：



## 7.5 closure

**章节回顾：**

* DLT：迭代最小化几何误差的一种方法
* 仿射相机中迭代最小化几何误差得到camera matrix
* 利用直线来去径向畸变

**计算camera matrix P的步骤汇总：**

* 给定5个世界到图像的correspondence point，对于一个零倾斜的相机矩阵P，一般有4个解，可以精确地将世界中的点映射到图像上

给定3个世界到图像的correspondence point对于已知标定K的相机矩阵P，一 般有4种解，可以精确地将世界映射到图像点。

* 如果想用线性算法来计算camera matrix P，则必须满足以下条件：

1. 已知道相机的位置（但不是方向）。
2. 照相机的principal ray的方向是已知的。
3. 相机的位置和相机的principal ray是已知的。
4. 相机的位置和相机的完整方向是已知的。
5. 相机的位置和方向是已知的，以及相机内部参数的一些子集（αx、αy、s、x0和y0）也是已知的。

**Principal axis和focal length重合：**

比较相机焦距∆f增加前后的深度d点的成像位置，或相机沿主轴向后位移∆t3的成像位置。

：表示沿着principal移动之后的坐标，表示沿着principal移动之前的坐标，其中，，

结论：估计的焦距参数和在主轴上的位置参数是相关的。

# 8 more single view geometry

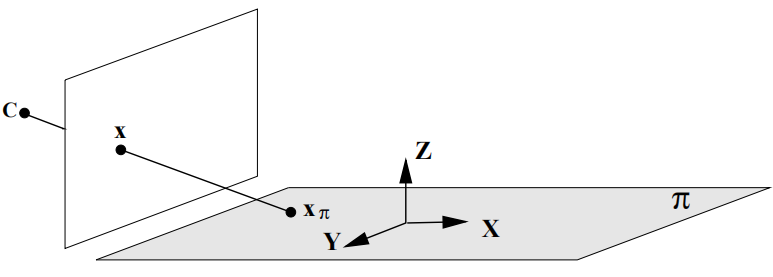
本章描述3D实体和图片之间在perspective projection之间的关系。

可以通过vanishing point计算出内参K

## 8.1 projective camera中的plane line conics

* **Plane**

：表示世界坐标系中的点X映射到图像中的x，我们选择世界坐标系的标准如下图



世界坐标系的XOY与与平面π重合，π是scene plane，C是camera，camera C不在scene plane上。Scene plane上的点X映射到image的x上

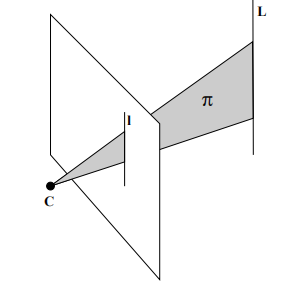
：scene plane 到image plane x的映射，H是3\*3的矩阵

：标定相机的目的就是得到P

：当世界坐标系的选定如上图所示的时候，就可以得到其中r1，r2为旋转矩阵的第一二列

* **Line**

：A，B是三维空间中的点，a,b是P下image上的两个点，表示3D世界中line上的一个点，



L:三维世界中的直线，C：camera centre l：image上的直线

从l到L叫back-project——反向投影

从L到l叫project——前向投影

：point x在l上

：得到x，x落在l上，空间中通过相机矩阵P映射到直线l的点的集合是平面。

：在camera mapping P下，将三维空的直线L映射到l

：是camera matrix P的每一行，表示有平面相交所形成的plucker line coordinate

:表示projection matrix，对line的效果就像P对point的效果，L表示3维世界中的plicker line coordinate

：a=pA b=PB

：

：表示line L经过camera centre

* **Conics**

：表示camera P下将圆锥体C反投影到cone上

：表示point x在conic上

:三维空间中 X在P下映射到image上的点x，也落在C上

：表示相机中心C是退化二次曲面的顶点

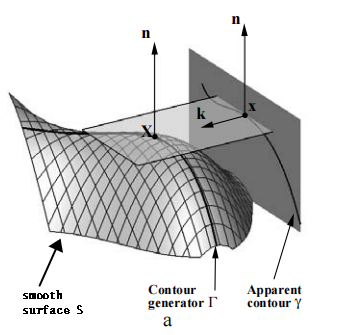
：已知,conic C被反投影到cone，camera centre 

个人总结：

已知camera matrix P的情况下，对三维空间中的point，plane，line，conic进行映射时，直接使用P进行计算，计算的公式都不一样，具体如下

|  |  |
| --- | --- |
| Point |  |
| Plane |  |
| Line |  |
| Conic |  |

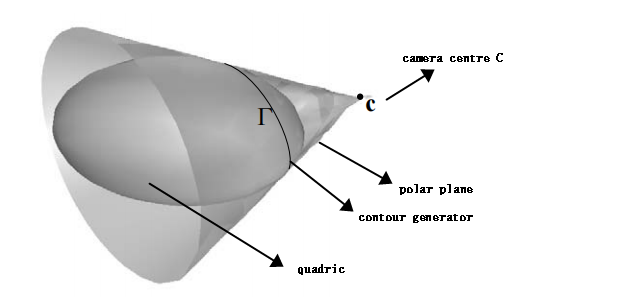
## 8.2 smooth surface的image

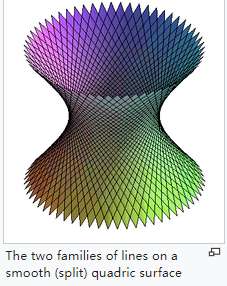


Contour generator：轮廓生成器

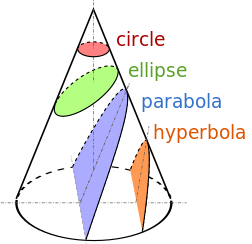
Apparent contour=outline=profile：表面轮廓，

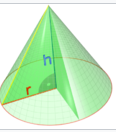
## 8.3 projective camera在quadric上的操作

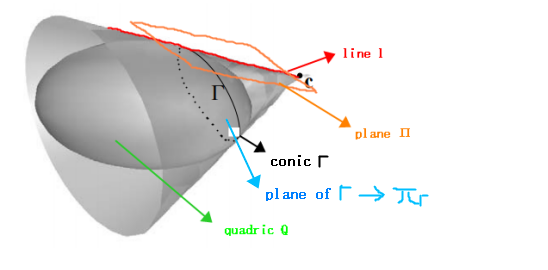


**Quadric：**二次曲面 ，可以是任何奇形怪状的曲面体

**Conic：**圆锥形，A conic section, conic or a quadratic curve is a curve obtained from a cone's surface intersecting a plane。相交可以得到双曲线（hyperbola），抛物线（parabola），椭圆（ellipse），本质是曲线



**Cone：**圆锥体



：将quadric Q通过camera matrix P转为conic C

：line l表示conic的切线

：line l通过P反投影得到平面Π

：平面Π是quadric Q的切平面

：

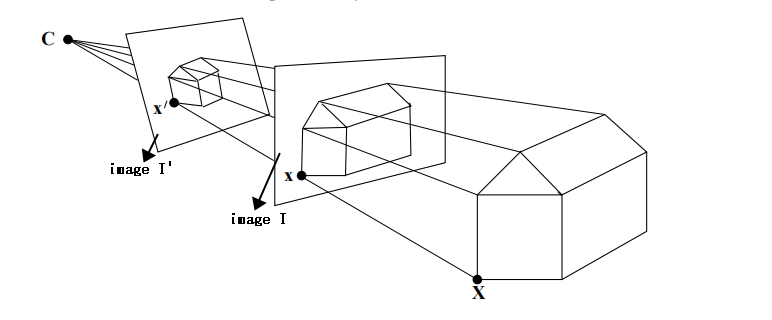
：Q为quadric ，C为camera centre，conic 可以用quadric表示

：quadric Q的切线和圆锥体的顶点一起构成degenerate quadric 

：quadric

：degenerate quadric

## 8.4 camera centre的重要性



：image I的camera matrix

：image I’的camera matrix

：因为I’和I的camera centre是同一个，所以他们的camera matrix存在关联

：根据camera matrix之间的关联推导出x和x’之间的关联

：表示世界坐标系和相机坐标系重合

### 8.4.1 移动image plane

一种移动image plane的方法是移动焦距

：三维空间中的点X到image中的点x的转换

：image I中的点x到image I’中的点x’的转换

：表示inhomogeneous principal point，表示放大因子，调整焦距的程度。

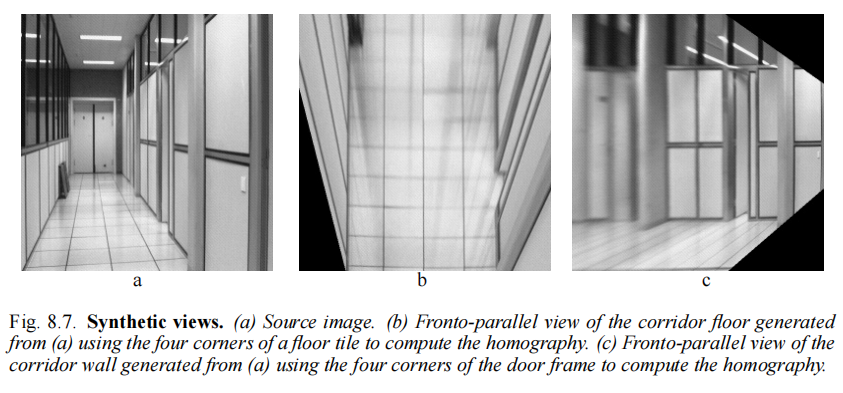
：根据相似三角形推导出

：在已知内参K的前提上，调整焦距k，可以推导出新的内参（而不用重新标定）

### 8.4.2 camera rotation

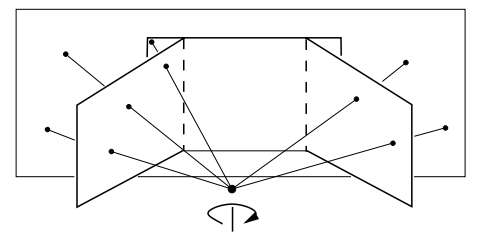
：

：H的真实特征值对应的特征向量是旋转轴的消失点。

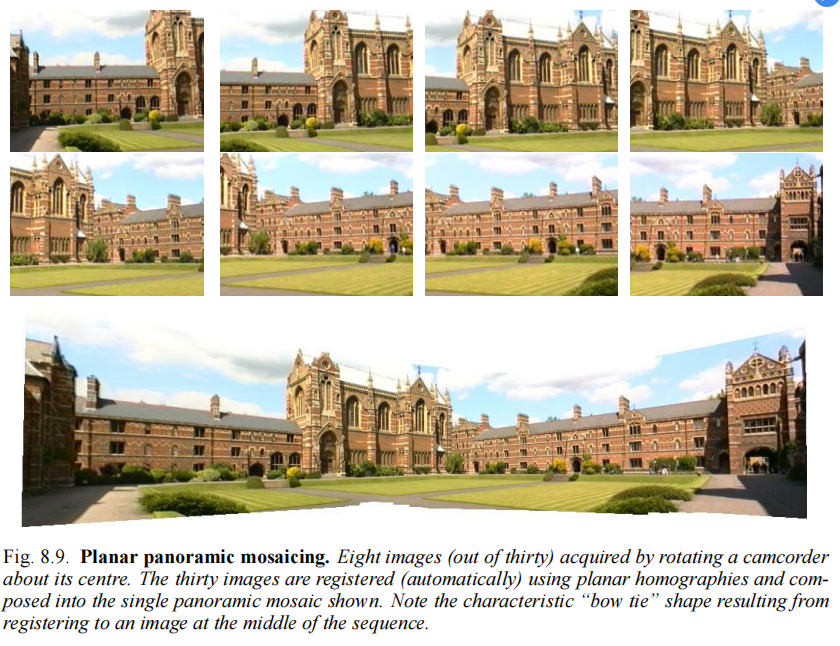


：与infinite homography相关

### 8.4.3 应用和例子



仅限于旋转相机的情况，如图所示，通过旋转相机获得的三个图像，选择中间的图像作为参考。计算另外两张图像到中间这张图像的homography H，用homography H扭曲图像，并用扭曲图像的不重叠部分来扩展参考图片的边缘部分。例如下图的例子（平面全景屏蔽）



### 8.4.4 projection notation

：world point

：image point

：camera matrix

：

:P的另一种写法

:camera matrix

### 8.4.5 move the camera centre

移动图像平面，同时固定相机中心，则图片之间的transformation只依赖于图像平面的运动，而不依赖于3维空间结构。如果相机中心被移动，image point的确定需要依赖于3维空间结构。

如何仅从图像判断相机中心是否已经移动？如果相机中心移动（不是沿着光线），图像重合就会丢失。

## 8.5 相机标定和图片的absolute conic

















8.5.1 图像的absolute conic

























8.5.2













