

FUNDAÇÃO GETÚLIO VARGAS
ESCOLA DE MATEMÁTICA APLICADA
CURSO DE GRADUAÇÃO EM MATEMÁTICA APLICADA

**GEOMETRIA PROJETIVA APLICADA
EM VISÃO COMPUTACIONAL**

VANESSA COSTA GAIO

Rio de Janeiro
2016

FUNDAÇÃO GETÚLIO VARGAS
ESCOLA DE MATEMÁTICA APLICADA
CURSO DE GRADUAÇÃO EM MATEMÁTICA APLICADA

**GEOMETRIA PROJETIVA APLICADA
EM VISÃO COMPUTACIONAL**

”Declaro ser o único autor do presente projeto de monografia que se refere ao plano de trabalho a ser executado para continuidade da monografia e ressalto que não recorri a qualquer forma de colaboração ou auxílio de terceiros para realizá-lo a não ser nos casos e para os fins autorizados pelo professor orientador”.

VANESSA COSTA GAIO

FUNDAÇÃO GETÚLIO VARGAS
ESCOLA DE MATEMÁTICA APLICADA
CURSO DE GRADUAÇÃO EM MATEMÁTICA APLICADA

**GEOMETRIA PROJETIVA APLICADA
EM VISÃO COMPUTACIONAL**

“Projeto de Monografia apresentado à Escola de
Matemática Aplicada como requisito parcial para
continuidade ao trabalho de monografia.”

Aprovado em _____ de _____ de _____

Professor Orientador: Luciano Monteiro de Castro

Sumário

1	Introdução	1
1.1	História da Geometria Projetiva	1
1.2	Espaços Projetivos	1
1.2.1	A Reta Projetiva Real: $\mathbb{P}^1(\mathbb{R})$	2
1.3	O Plano Projetivo Real: $\mathbb{P}^2(\mathbb{R})$	6
1.4	Retas em \mathbb{P}^2	6
1.5	Dualidade	7
2	Desenvolvimento	8
2.1	Calibração	8
2.1.1	Transformações de Câmera	8
2.2	Método de Tsai	12
2.2.1	Implementação do Método de Tsai	15
2.2.2	Exemplos de Aplicação do Método de Tsai	17
2.3	Método de Zhang	18
2.3.1	Implementação do Método de Zhang	23
2.3.2	Exemplo de Aplicação do Método de Zhang	27
3	Conclusão	28
4	Referências Bibliográficas	29

1 Introdução

Iniciaremos este TCC com o desenvolvimento dos conceitos da disciplina de Geometria Projetiva, abordando tópicos como seus fundamentos, princípio de dualidade, projetividades. E com isso aplicá-lá em conceitos de visão computacional.

1.1 História da Geometria Projetiva

A Geometria Projetiva pode se definir como a Geometria na qual se reúnem as propriedades que permanecem invariantes quando se realiza uma projeção a partir de um ponto. Duas figuras geométricas são consideradas equivalentes se podemos obter uma a partir da outra através de uma projeção. Além disso, a Geometria Projetiva é a geometria "só da régua", diferente da Geometria Euclidiana que é a geometria da "régua" e do "compasso". Isso porque, basicamente, seus objetos envolvem a colinearidade (de pontos) e a incidência (de retas), dois critérios que são invariantes segundo projeções (sobre um plano). Nesta geometria os axiomas são construídos de modo que não se admita retas paralelas, ou seja, toda reta no plano se cruza, sem exceção alguma. Esse fato faz com que a teoria ganhe uma certa simetria.

A necessidade da criação deste tipo de geometria surgiu na Itália no século XV, junto com o Renascimento, devido à dificuldade dos artistas em buscar mais realismo para suas obras. Com isso, começaram a introduzir os conceitos de ponto de fuga e perspectiva, de modo que qualquer pessoa identificasse sem dificuldade o que estava perto e o que estava longe. Um nome renascentista, já no século XVII, se debruça sobre a mesma tarefa e realiza resultados que mais se ligam à Geometria Projetiva do que à Perspectiva dos artistas. Girard Desargues, arquiteto, mesmo tendo em vista suprir necessidades imediatas de sua profissão, em seus estudos acaba por produzir teoremas e eixos conceituais que o tornam o primeiro sistematizador do pensamento projetivo. Ainda no século XVII, Blaise Pascal, que foi contemporâneo de Desargues, ao debruçar-se sobre os estudos deste, certamente com acesso às mais recentes descobertas científicas, investiga as cônicas e suas propriedades projetivas.

Já no século XIX, Jean Victor Poncelet foi o primeiro a reconhecer a geometria Projetiva como um novo ramo da matemática e se propôs a achar todas as propriedades geométricas das figuras que são invariantes por projeções e seções. O trabalho de Poncelet baseia-se sobre três idéias básicas. Estas idéias, na realidade, são anteriores a Poncelet, e apenas tomou como base para o seu trabalho dando a elas uma ampla aplicação. A primeira idéia usa a noção de figuras homólogas (uma figura pode ser obtida da outra, por uma sequência de projeções e seções), com o objetivo de encontrar uma figura mais simples do que a original, e estudar as propriedades que são invariantes obtendo propriedades da figura mais complexa. A segunda idéia é o princípio de continuidade, nas palavras de Poncelet o princípio é: "Se uma figura é obtida de outra por mudança contínua, e a última é tão geral quanto a primeira, então qualquer propriedade da primeira pode ser enunciada imediatamente para a segunda figura", para que o princípio funcione em várias situações, Poncelet teve que introduzir, além dos pontos no infinito do espaço projetivo, a noção de pontos imaginários. A terceira idéia é a da reciprocidade polar em relação a uma cônica, com o objetivo de tentar dar uma prova do princípio de dualidade.

1.2 Espaços Projetivos

A ideia de um espaço projetivo se refere a perspectiva, ou seja, à maneira como um olho ou uma câmera projeta uma cena que está 3D em uma imagem 2D. Todos os pontos que estão em uma linha de projeção, cruzando com a pupila de entrada da câmera, são projetados em um ponto de imagem comum. Neste exemplo, o espaço vectorial é \mathbb{R}^3 com a pupila de entrada da câmara na origem, e o espaço projetivo corresponde aos pontos da imagem. Ou seja, o espaço projetivo é um objeto geométrico que formaliza declarações como "linhas paralelas se cruzam

no infinito.”

Definição: $\mathbb{P}(V)$, o espaço projetivo gerado por V (espaço vetorial) é o espaço quociente $(V - \{0\})/\sim$, onde \sim é a seguinte relação de equivalência.

$$u \sim v \iff u = \lambda v, \text{ para algum escalar } \lambda \in \mathbb{K} \text{ (corpo)}$$

Chamando os elementos de $\mathbb{P}(V)$ de pontos, temos que cada ponto de $\mathbb{P}(V)$ é uma classe de equivalência de vetores de V pela relação \sim . Dado $v \in (V - \{0\})$ a classe de v ($[v]$) é o conjunto de todos os vetores não nulos que são múltiplos de, ou paralelos a v .

Podemos considerar $\mathbb{P}(V)$ como o conjunto de todos os subespaços de V de dimensão 1. Quando V tem dimensão finita $n + 1$, diremos que $\mathbb{P}(V)$ tem dimensão n , e devido ao isomorfismo entre V e \mathbb{K}^{n+1} , podemos identificar $\mathbb{P}(V)$ com o espaço $\mathbb{P}(\mathbb{K}^{n+1})$ (ou com $\mathbb{P}^n(\mathbb{K})$ espaço projetivo n -dimensional sobre \mathbb{K}).

1.2.1 A Reta Projetiva Real: $\mathbb{P}^1(\mathbb{R})$

Este trabalho irá utilizar espaços projetivos reais. Para este caso de reta projetiva real, cada ponto de \mathbb{P}^1 é uma classe de equivalência de vetores de $\mathbb{R}^2 - \{(0, 0)\}$ por \sim , ou seja um conjunto de vetores paralelos (e não nulos) no plano real. Com isso, cada ponto de \mathbb{P}^1 pode ser visto como uma reta de \mathbb{R}^2 passando pela origem.

• Coordenadas Projetivas ou Coordenadas Homogêneas

Na geometria computacional as coordenadas homogêneas são utilizadas no lugar do sistema de coordenadas cartesianas devido às vantagens que oferecem no tratamento algébrico de pontos "no infinito".

Definição: Um ponto representado por (X, Y) no sistema de coordenadas cartesiano é representado, em coordenadas homogêneas, por $[x, y, t]$, onde $X = \frac{x}{t}$, $Y = \frac{y}{t}$ e t é chamado de peso, para t não zero. Assim, o ponto cartesiano (X, Y) corresponde à uma infinidade de triplas $[tX, tY, t]$, incluindo o caso particular $[x, y, 1]$.

Este sistema permite representar pontos no infinito (ou direções), chamados de pontos impróprios, ao usar $t = 0$ e $x \neq 0$ ou $y \neq 0$.

Escolhida uma base para \mathbb{R}^2 , cada vetor não nulo $v \in \mathbb{R}^2$ possui coordenadas (v_1, v_2) , não ambas nulas. Existe um único ponto $P \in \mathbb{P}$ correspondente à classe de equivalência de $[v]$. Assim $[v_1, v_2]$ são as coordenadas projetivas de P . As coordenadas projetivas também são chamadas de coordenadas homogêneas, devido à propriedade de continuarem representando o mesmo ponto ao serem multiplicadas por um escalar não nulo, o que decorre diretamente da definição de espaço projetivo.

• Relação com a Reta Real

No $V = \mathbb{R}^2$ temos a chamada reta projetiva. Quando estudamos um espaço quociente sob uma relação de equivalência, é comum escolhermos um único representante para cada classe. No caso do estudo de \mathbb{P}^1 devemos escolher um representante para cada classe de equivalência como uma reta contida em \mathbb{R}^2 que não passe pela origem, sendo que esta reta corta cada ponto da reta projetiva uma única vez (os pontos de \mathbb{R}^2 são vetores com origem em $(0, 0)$ e extremidade no ponto correspondente).

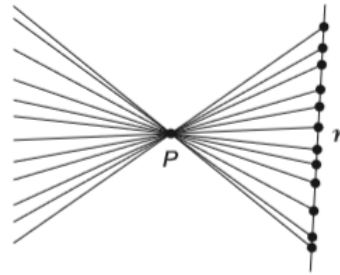
Exemplo: Seja a reta $r : y = 1$ de \mathbb{R}^2 , cada ponto $[X, Y] \in \mathbb{P}^1$ com $Y \neq 0$ corta r no ponto $(\frac{X}{Y}, 1)$, que é o único representante de $[X, Y]$ em r . Reciprocamente, todo ponto $(x, 1)$ de r pertence a uma única classe de equivalência $[x, 1] \in \mathbb{P}^1$. Se $Y = 0$, o ponto $[X, Y] = [X, 0]$, não possui representante sobre r . Existe apenas um ponto de \mathbb{P}^1 nesta situação, já que como X e Y não podem ser ambos nulos, temos $X \neq 0$ e portanto $[X, 0] = [\frac{X}{X}, \frac{0}{X}] = [1, 0]$.

$$\mathbb{P}^1(\mathbb{R}) = \{[x, 1] | x \in \mathbb{R}\} \cup \{[0, 1]\}$$

A reta projetiva real pode ser visualizada como o resultado de se acrescentar um ponto à reta real. Tal ponto é muitas vezes denominado o ponto do infinito. De fato, observamos que quando fazemos um ponto r tender ao infinito, em ambos os sentidos, a reta que representa a classe de equivalência em \mathbb{P}^1 vai se aproximando da reta $y = 0$, classe do ponto do infinito $[0, 1]$.

• Teorema Fundamental da Geometria Projetiva

É conveniente na geometria projetiva pensarmos em pontos como feixes de retas, e em retas como fileiras de pontos. Assim, veremos que um feixe de retas P projeta uma fileira de pontos sobre uma reta, ou ainda, a fileira de pontos r é uma seção do feixe.

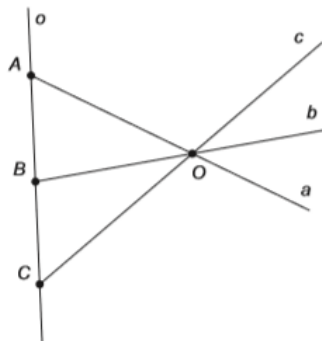


Dizemos que o ponto X projeta a reta x se $X = x.o$, onde x passa por O distinto de X e o é uma reta não incidente a O e diferente da reta x . Neste caso dizemos também que a reta x projeta o ponto X . Chamaremos esse fato de correspondência elementar e denotaremos da seguinte forma:

$$X \bar{\wedge} x \text{ ou } x \bar{\wedge} X$$

Se $A \bar{\wedge} a$, $B \bar{\wedge} b$ e $C \bar{\wedge} c$ com A , B e C incidentes à o e com a, b, c passando por O , podemos escrever:

$$ABC \bar{\wedge} abc \text{ ou } abc \bar{\wedge} ABC$$



Nesse caso dizemos que a fileira ABC é uma seção do feixe abc pelo ponto O .

Chamaremos de projetividade uma combinação finita de correspondências elementares:

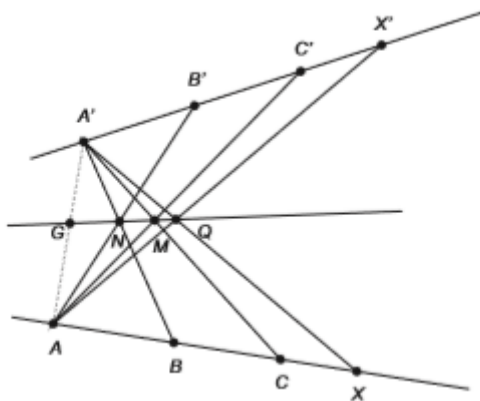
$$X\bar{\wedge}x^n \text{ ou } x\bar{\wedge}X^n \text{ ou } X\bar{\wedge}X^n \text{ ou } x\bar{\wedge}x^n$$

Teorema: Uma projetividade é unicamente determinada quando são dados três pontos colineares e seus respectivos correspondentes.

Demonstração: Dados A, B e C colineares e A', B' e C' , seus respectivos correspondentes em uma projetividade qualquer, deveremos determinar um ponto X' de modo que $ABCX\bar{\wedge}A'B'C'X'$ para um ponto X qualquer na reta AB . Para isso, vamos dividir em dois casos: quando os pontos A, B e C estão em uma reta distinta de A', B' e C' , sendo que a interseção dessas retas não é nenhum desses pontos e quando isso não acontece.

Caso 1: Os pontos A, B e C estão em uma reta distinta de A', B' e C' , sendo que a interseção dessas retas não é nenhum desses pontos.

Seja $N = AB'.BA'$ e $M = AC'.CA'$, as retas AA' e XA' interceptam a reta MN em dois pontos distintos, G e Q respectivamente.



Assim o ponto X' fica determinado pela interseção das retas AQ e $A'B'$. E isso é verdade pois:

$$AA'.MN = G$$

$$BA'.MN = N$$

$$CA'.MN = M$$

$$XA'.MN = Q$$

Então, $ABCX\bar{\wedge}GNMQ$

$$GA.A'B' = A'; \text{ pois } GA = GA' \text{ já que } G = AA'.MN, \text{ daí, } GA.A'B' = GA'.A'B' = A'.$$

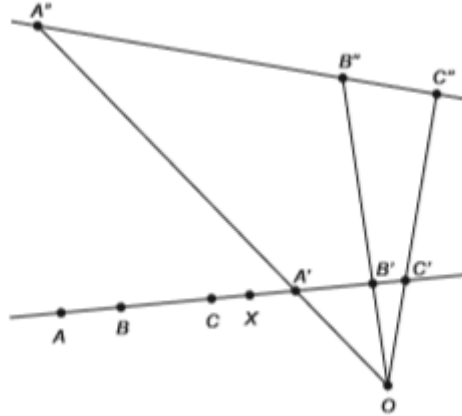
$$NA.A'B' = B'; \text{ pois } NA = NB' \text{ já que } N = BA'.AB', \text{ daí, } NA.A'B' = NB'.A'B' = B'.$$

$$MA.A'B' = C'; \text{ pois } MA = MC' \text{ já que } M = CA'.AC', \text{ daí, como } A', B' \text{ e } C' \text{ são colineares, } MA.A'B' = MC'.A'C' = C'$$

$$QA.A'B' = X'; \text{ pois } QA = QX' \text{ e } A'B' = A'X' \text{ já que } X' = AQ.A'B', \text{ daí, } QA.A'B' = QX'.A'X' = X'.$$

Logo $GNMQ \bar{\wedge} A'B'C'X'$. Conclusão $ABCX \bar{\wedge} A'B'C'X'$

Caso 2: Se A, B, C, A', B' e C' estiverem na mesma reta, por exemplo.



Pela construção teremos $A'B'C' \bar{\wedge} A''B''C''$. Usando o caso 1 para os pontos A, B, C, X, A'', B'' e C'' ficamos com $ABCX \bar{\wedge} A''B''C''X''$. Desses dois resultados temos $ABCX \bar{\wedge} A'B'C'X'$ onde X' é o ponto que OX'' intercepta a reta $A'B'$.

Provando agora que X' é único, vamos supor que exista X'' tal que:

$$ABCX \bar{\wedge} A'B'C'X' \text{ e } ABCX \bar{\wedge} A'B'C'X''$$

Concluimos que $A'B'C'X' \bar{\wedge} ABCX \bar{\wedge} A'B'C'X''$, ou ainda $A'B'C'X' \bar{\wedge} A'B'C'X''$. Como esta projetividade deixa invariante os três pontos A', B' e C' , e sabendo que se uma projetividade deixa invariante três pontos distintos de uma fileira, então ela deixa invariante toda a fileira, $X' = X''$, ou seja X' é único.

Caso particular do teorema: Sejam A, B e C três pontos distintos de \mathbb{P}^1 . É sempre possível escolher uma base de \mathbb{R}^2 para qual as coordenadas projetivas desses pontos sejam:

$$A = [1, 0], B = [0, 1], C = [1, 1]$$

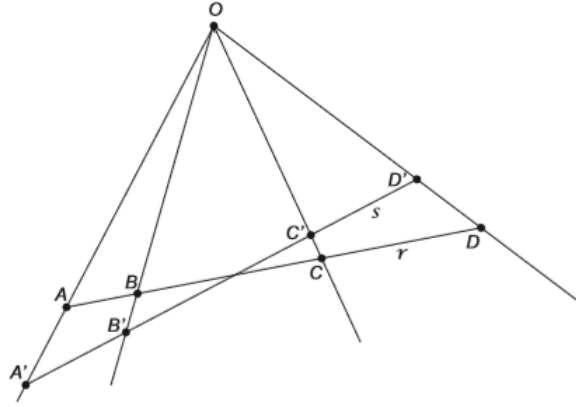
Demonstração: Sejam v_A, v_B e v_C representantes quaisquer das classes de equivalência de A, B e C em $\mathbb{R}^2 - \{0\}$. Como $A \neq B$, v_A e v_B não são paralelos, logo são independentes e formam uma base de \mathbb{R}^2 . Seja (a, b) a representação de v_C nesta base. Como C é distinto de A e de B , a e b são não nulos, pois v_C não pode ser paralelo a v_A ou v_B . Escolhendo a base (av_A, bv_B) para \mathbb{R}^2 , temos $A = [v_A] = [av_A] = [1, 0]$, $B = [v_B] = [bv_B] = [0, 1]$ e $C = [v_C] = [av_A + bv_B] = [1, 1]$.

Com isso podemos falar de coordenadas projetivas sem a precisar voltar ao espaço gerador \mathbb{R}^2 . A, B e C formam uma referência projetiva de \mathbb{P}^1 . A base de \mathbb{R}^2 cuja existência provada no teorema não é única. Mas as coordenadas projetivas induzidas por ela o são.

• Razão Cruzada

Uma transformação projetiva preserva a razão cruzada, ou seja, essa razão é invariante e diz que se quatro retas por um ponto O interceptam uma reta r nos pontos A, B, C e D e uma reta s nos pontos A', B', C' e D' (ou se A', B', C' e D' em s são projeções de A, B, C e D em r por um ponto O), então, para toda r e s não incidentes à O temos:

$$\frac{CA/CB}{DA/DB} = \frac{C'A'/C'B'}{D'A'/D'B'}$$



1.3 O Plano Projetivo Real: $\mathbb{P}^2(\mathbb{R})$

Nossos pontos continuam sendo retas passando pela origem, porém agora as retas estão contidas em \mathbb{R}^3 . Podemos escolher um representante para cada classe de equivalência cortando essas retas pelo plano $z = 1$ em \mathbb{R}^3 . Mas, com isso, perdemos todos os pontos do plano projetivo correspondentes a retas paralelas ao plano $z = 1$. Este conjunto de retas tem exatamente a mesma estrutura do que o conjunto de classes de equivalência que define a reta projetiva, portanto faz sentido imaginar que agora temos uma reta de pontos no infinito. Ou seja, \mathbb{P}^2 é obtido acrescentando uma reta do infinito ao plano real.

Para cada ponto $P = [X, Y, Z]$ do plano projetivo há duas possibilidades:

(i) Se $Z \neq 0$, sendo $x = \frac{X}{Z}$ e $y = \frac{Y}{Z}$ temos $[X, Y, Z] = [x, y, 1]$, de forma que o ponto P possui um único representante do plano real $z = 1$. O mesmo ocorre se pensarmos que cada ponto $(x, y, 1)$ do plano $z = 1$ corresponde a um único ponto P com $Z \neq 0$. Ou seja, existe uma bijeção entre \mathbb{R}^3 e os pontos de \mathbb{P}^2 com coordenada projetiva Z não nula.

(ii) Se $Z = 0$, ao ponto $P = [X, Y, 0]$ associamos bijetivamente o ponto $[X, Y] \in \mathbb{P}^1$, logo existe uma bijeção natural entre pontos que não têm representante no plano real $z = 1$, os chamados pontos no infinito e a reta projetiva real. Isso justifica chamarmos este conjunto de pontos de reta do infinito.

1.4 Retas em \mathbb{P}^2

A reta do infinito é um subconjunto (retas em \mathbb{P}^2) de \mathbb{P}^2 que é formado por retas coplanares de \mathbb{R}^3 passando pela origem.

Lema: (Condição de alinhamento de três pontos) Os pontos A , B e C do plano projetivo estão alinhados se, e somente se, é nulo o determinante da matriz 3×3 cujas linhas são as coordenadas desses pontos em uma determinada referência projetiva.

Corolário: As retas do plano projetivo são os conjuntos solução das equações homogêneas de primeiro grau

$$aX + bY + cZ = 0$$

Onde $[X, Y, Z]$ representam as coordenadas homogêneas de um ponto genérico de \mathbb{P}^2 e a , b , c são constantes reais não todas nulas.

Demonstração: O ponto $[X, Y, Z]$ pertence a reta definida por $A = [A_1, A_2, A_3]$ e $B = [B_1, B_2, B_3]$, com A e B distintos, se e somente se

$$\begin{vmatrix} A_1 & A_2 & A_3 \\ B_1 & B_2 & B_3 \\ X & Y & Z \end{vmatrix} = 0 \iff \begin{vmatrix} A_2 & A_3 \\ B_2 & B_3 \end{vmatrix} X + \begin{vmatrix} A_3 & A_1 \\ B_3 & B_1 \end{vmatrix} Y + \begin{vmatrix} A_1 & A_2 \\ B_1 & B_2 \end{vmatrix} Z = 0$$

A última equação é homogênea de primeiro grau em X, Y, Z , e seus coeficientes são não nulos, pois $A \neq B$. Reciprocamente, supondo $a \neq 0$, as soluções de $aX + bY + cZ = 0$ são os pontos da reta que é definida por $[b, -a, 0]$ e $[c, 0, -a]$.

Teorema: (Teorema Fundamental aplicado ao Plano Projetivo) Sejam A, B, C, D quatro pontos distintos do plano projetivo tais que quaisquer três entre eles não pertencem a uma mesma reta. É sempre possível escolher uma base de \mathbb{R}^3 para a qual as coordenadas projetivas desses pontos sejam

$$A = [1, 0, 0], B = [0, 1, 0], C = [0, 0, 1], D = [1, 1, 1]$$

Teorema de Desargues: Sejam ABC e $A'B'C'$ dois triângulos no plano plano projetivo tais que as retas que unem vértices correspondentes são concorrentes, ou seja, existe um ponto O pertencente às retas AA', BB' e CC' . Então os lados correspondentes (AB e $A'B'$, BC e $B'C'$, CA e $C'A'$) intersectam-se em pontos colineares.

Demonstração: Vamos escolher $(A, B, C; O)$ como referência projetiva, a reta OA tem como equação $Y = Z$ (pois O e A são soluções dessa equação). A é o único ponto desta reta com coordenadas Y e Z nulas, logo A' possui coordenadas $[a, 1, 1]$ com $a \in \mathbb{R}$. Analogamente, $B' = [1, b, 1]$, $C' = [1, 1, c]$ com $b, c \in \mathbb{R}$. A reta BC tem equação $X = 0$, logo o ponto P de interseção desta reta com $B'C'$ tem coordenadas $P = [0, P_2, P_3]$ tais que

$$\begin{vmatrix} 1 & b & 1 \\ 1 & 1 & c \\ 0 & P_2 & P_3 \end{vmatrix} = 0 \iff (1 - c)P_2 + (1 - b)P_3 = 0$$

Assim $P = [0, b - 1, 1 - c]$. Analogamente $Q = [1 - a, 0, c - 1]$ e $R = [a - 1, 1 - b, 0]$, onde Q é a interseção de CA com $C'A'$ e R é a interseção de AB com $A'B'$. Logo temos:

$$\begin{vmatrix} 0 & b - 1 & 1 - c \\ 1 - a & 0 & c - 1 \\ a - 1 & 1 - b & 0 \end{vmatrix} = 0$$

Ou seja, P, Q e R são colineares, pois a soma das linhas do determinante acima é nula.

1.5 Dualidade

Nas frases "cada dois pontos determinam uma única reta" e "cada duas retas determinam um único ponto", se trocarmos na primeira afirmação a palavra "ponto" pela palavra "reta" (respeitando a concordância verbal quando necessário), obtemos a segunda afirmação e vice-versa.

Esse fenômeno é mais que uma casualidade. Na geometria projetiva, tudo que acontece com um ponto, pode acontecer também, igualmente, com uma reta. Podemos usar dessa simetria para fazer novas construções. Veja o caso dos quadrângulos e quadriláteros, as definições são

praticamente idênticas se ignorarmos as palavras ponto e reta (e todas as outras que fazem referência a eles, por exemplo, colineares e concorrentes). Quando isso acontece entre duas figuras, dizemos que elas são duais. Dessa forma o dual de um quadrângulo é um quadrilátero, assim como, o dual de três retas concorrentes são três pontos colineares. Para cada construção, podemos achar uma dual, e as vezes elas coincidem. No triângulo por exemplo, são três retas determinando três pontos ou três pontos determinando três retas. O triângulo, por isso, é uma figura auto-dual.

A grande vantagem desse princípio é que podemos mecanicamente dualizar os enunciados e as provas de teoremas afim de obter novos resultados sem mínimos esforços. É comum representar um ponto por uma letra maiúscula e sua reta dual pela mesma letra, mas minúscula, e vice-versa. Outra notação comum é acrescentar um $(*)$ ao símbolo de um ponto ou reta para representar seu dual.

2 Desenvolvimento

A Geometria Computacional é o ramo da ciência da computação que estuda as técnicas e algoritmos para resolução computacional de problemas geométricos. Surgiu por volta de 1970, quando houve um grande progresso da computação gráfica e robótica, para as quais é essencial resolver problemas geométricos de forma eficiente. Juntas, a geometria projetiva e a geometria computacional contribuem para melhoria de inúmeras aplicações em diferentes áreas, como Processamento de Imagens, Visão Computacional, Sistemas de Informações Geográficas, entre outras. Neste tópico iremos relacionar a Geometria Projetiva com a Visão Computacional, olhando para o mundo da fotografia 3D e abordando assuntos como calibração de câmeras, para com isso termos acesso de onde foi tirada determinada fotografia.

2.1 Calibração

A calibração de uma câmara de imagem é um processo que permite estimar os parâmetros que definem a projeção de um ponto tridimensional ($3D$) do mundo num ponto bidimensional ($2D$) da imagem. Os referidos parâmetros incluem a geometria interna e óptica da câmara, assim como o seu posicionamento e orientação no espaço $3D$. A utilização de câmaras de imagem devidamente calibradas possibilita a resolução de problemas computacionais relacionados com a obtenção da posição e orientação $3D$ (pose).

2.1.1 Transformações de Câmera

Iremos obter aqui uma expressão matemática da função que associa, a cada ponto do espaço, expresso em um sistema de referência ortogonal, a posição correspondente na imagem capturada por uma câmara (ou seja, a transformação associada à projeção de pontos $3D$ do mundo, de coordenadas (X, Y, Z) , em pontos $2D$ da imagem, de coordenadas (u, v)). Mas, para escrever esta função precisamos ter informações sobre a câmara; ou seja, precisamos especificar certos parâmetros, que definem seu comportamento. Os parâmetros extrínsecos fornecem a posição e orientação da câmara em causa, definidas por intermédio de uma matriz de rotação R e um vetor de translação T , em relação a um certo sistema de coordenadas $3D$ global (mundo). Os parâmetros intrínsecos representam as características ópticas e geométricas internas da câmara (como a distância focal f , fatores de escala (s_x, s_y) , posição em pixels da projeção ortogonal do centro óptico no plano de projeção (u_c, v_c) e as imperfeições geométricas introduzidas pelo sistema sensor da câmara τ).

Para definir a correspondência dos pontos precisamos primeiramente definir quatro sistemas de coordenadas e assim relacioná-las para obter a expressão matemática desejada. Os sistemas

de coordenadas são o sistema de coordenadas do mundo (SCM), sistema de coordenadas da câmera (SCC), sistema de coordenadas da imagem (SCI) e sistema de coordenadas em pixel (SCP).

- **Sistema de Coordenadas do Mundo (SCM):** Sistema tridimensional, escolhido de forma a ser conveniente para a descrição da cena ou para objetos utilizados para calibrar as câmeras. O SCM tem suas coordenadas denominada por (X, Y, Z) .

- **Sistema de Coordenadas da Câmera (SCC):** Sistema tridimensional com origem no centro óptico da câmera, os eixos X e Y são paralelos a borda da imagem a ser formada sobre o plano de projeção e o eixo Z é perpendicular a imagem. O SCC tem suas coordenadas denominadas por $(\tilde{X}, \tilde{Y}, \tilde{Z})$. Além disso chamamos de f a distância focal da câmera (distância entre o centro óptico da câmera e o plano de projeção), assim a equação que representa o plano de projeção é dada por $\tilde{Z} = f$.

- **Sistema de Coordenadas da Imagem (SCI):** Sistema bidimensional localizado no plano de projeção. Sua origem é a projeção ortogonal do centro óptico da câmera. O SCI tem suas coordenadas denomida por (x, y) .

- **Sistema de Coordenadas em Pixel (SCP):** Sistema bidimensional, com coordenadas em pixels (menor ponto que forma uma imagem digital, sendo que o conjunto de milhares de pixels formam a imagem inteira), que define a posição de um ponto da imagem com relação à grade de pixels. A origem costuma está localizada no cando superior ou inferior da imagem. O SCP tem suas coordenadas denomida por (u, v) .

A partir de agora iremos relacionar o SCM com o SCC, o SCC com o SCI e o SCI com o SCP, pois o imageamento de um ponto do espaço através de uma câmera digital consiste em uma sequência de transformações entre estes espaços.

- **Do SCM para o SCC:** Consiste em realizar a mudança de coordenadas entre dois referenciais ortogonais do espaço tridimensional. Sendo T um vetor de translação que fornece a origem O do SCM no referencial SCC e R uma matriz de rotação com colunas r_1, r_2 e r_3 que são as coordenadas dos vetores i, j e k do SCM com relação à base formada pelos vetores \tilde{i}, \tilde{j} e \tilde{k} do SCC. Assim, temos que dado um ponto $P = (X, Y, Z)$ do SCC, suas coordenadas no SCC são as do vetor $\tilde{OP} = \tilde{OO} + OP$, ou seja:

$$(\tilde{X}, \tilde{Y}, \tilde{Z}) = T + Xr_1 + Yr_2 + Zr_3$$

Assim, sendo $\tilde{P} = (\tilde{X}, \tilde{Y}, \tilde{Z})$, temos:

$$\tilde{P} = RP + T$$

Em coordenadas homogêneas:

$$\begin{bmatrix} \tilde{X} \\ \tilde{Y} \\ \tilde{Z} \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Como os referenciais SCM e SCC são ortogonais, a matriz R é ortogonal ($RR^t = I$). Assim a transformação inversa será representada por:

$$P = R^t \tilde{P} - R^t T$$

Como já foi dito anteriormente R e T representam os parâmetros extrínsecos da câmera.

• **Do SCC para o SCI:** Em câmeras pinhole (máquina fotográfica sem lente), consiste em uma projeção perspectiva (o caso em que há lentes será visto a diante) Assim, sendo f a distância focal a transformação será dada por:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \simeq \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \tilde{X} \\ \tilde{Y} \\ \tilde{Z} \\ 1 \end{bmatrix}$$

Repare que a transformação acima é não invertível, pois cada ponto do espaço corresponde a um único ponto da imagem, mas um ponto da imagem corresponde a infinitos ponto no espaço.

• **Do SCI para o SCP:** Sabemos que um raio luminoso que atinge o plano de formação da imagem, é registrado por um sensor que usualmente estão dispostos segundo uma matriz retangular. Porém pode ocorrer das linhas não terem o mesmo espaçamento das colunas ou as linhas não serem perpendiculares às colunas. Assim este fato pode ser modelado colocando o eixo horizontal do SCI correspondendo exatamente ao eixo horizontal do SCP e os eixos verticais não necessariamente alinhados. Além disso, sabemos que a origem do SCI encontra-se na projeção do centro óptico (em alguns casos centro da imagem) e a origem do SCP encontra-se em um dos cantos da imagem. Logo a transformação será dado por:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & \tau & u_c \\ 0 & s_y & v_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Como dito anteriormente, assim como o f , os parâmetros s_x, s_y, u_c, v_c e τ são os parâmetros intrínsecos da câmera. Onde s_x e s_y representam o número de pixels por unidade de comprimento respectivamente nas direções horizontal e vertical. u_c e v_c fornecem a posição em pixels da projeção do centro óptico da câmera. E τ é a tangente do ângulo formado entre a coluna vertical de pixels e a perpendicular às linhas de pixels.

Finalmente podemos levar um ponto do SCC para o SCP compondo as transformações acima:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \simeq \begin{bmatrix} s_x & \tau & u_c \\ 0 & s_y & v_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Sendo $[p]$ a representação das coordenadas homogêneas do ponto no SCP e $[P]$ no SCM temos:

$$[p] \simeq \begin{bmatrix} f s_x & f \tau & u_c \\ 0 & f s_y & v_c \\ 0 & 0 & 1 \end{bmatrix} [R \ T] [P]$$

Ou ainda,

$$[p] \simeq K [R \ T] [P]$$

Onde K é a matriz de calibração que reúne os parâmetros intrínsecos da câmera e $\begin{bmatrix} R & T \end{bmatrix}$ representa os parâmetros extrínsecos.

Sabemos que o procedimento de calibração consiste em estimar os parâmetros a partir de pares de pontos do espaço e suas respectivas posições na imagem e a matriz K tem uma consequência importante neste procedimento. Como na matriz K temos representado os produtos $f s_x$ e $f s_y$, não é possível estimar separadamente cada um, somente se for dado informações sobre os valores de s_x e s_y . Assim podemos escrever a matriz K como:

$$K = \begin{bmatrix} f_x & c & u_c \\ 0 & f_y & v_c \\ 0 & 0 & 1 \end{bmatrix}$$

Como foi dito anteriormente, veremos agora como tratar dos casos em que a câmera possui lente, o que não ocorre nas câmeras pinhole.

• Modelando as Lentes

Em Fotografia 3D não levar em consideração o desvio dos raios luminosos causados pelas lentes pode levar a imprecisões inaceitáveis na reconstrução tridimensional. Porém a modelagem exata do comportamento das lentes não é prática, por causa da sua complexidade.

Uma alternativa é a utilização de um modelo que considera apenas a distorção radial, que afeta de forma similar todos os pontos da imagem que equidistam do centro. Essa distorção afeta a transformação do SCC para SCI. Ou seja, a imagem deixa de ser formada no ponto (x, y) para ser formada no ponto (x', y') . Assim temos:

$$(x, y) = (1 + d)(x', y')$$

Onde d depende de $r = \sqrt{(x')^2 + (y')^2}$, usualmente pela seguinte função polinomial $d = k_1 r^2 + k_2 r^4$, com k_1 e k_2 sendo os coeficientes de distorção radial. Porém em casos que o ângulo de visão da lente não é muito grande podemos levar em consideração somente o coeficiente k_1 .

Calibrar uma câmera consiste em estimar os parâmetros intrínsecos e extrínsecos, e para isso precisamos obter um conjunto de pontos na imagem $(p_1, p_1, ..p_n)$ que correspondem a um conjunto de pontos no espaço tridimensional $(P_1, P_2, ..., P_n)$, assim obtemos os parâmetros intrínsecos tais que as imagens $\hat{p}_1, \hat{p}_2, ..., \hat{p}_n$ obtida com uma câmera utilizando estes parâmetros estejam o mais próximo possível das imagens observadas. Com isso formular um problema de calibração consiste em minimizar $\sum_{i=0}^n ||\hat{p}_i - p_i||^2$ para todos os possíveis valores de R , T e K .

Algumas possibilidades para formular o problema através de um modelo simplificado são:

- Ignorar a distorção radial, considerando um modelo de câmera sem lente.
- Considerar a projeção do centro óptico da câmera no centro da imagem.
- Considerar os pixels quadrados, ou seja, $s_x = s_y$, então $f_x = f_y$.
- Considerar as colunas de pixels ortogonais, ou seja, $\tau = 0$ então $c = 0$

Os métodos de calibração são usualmente classificados em dois grandes grupos:

• **Calibração Tradicional:** estes métodos associam pontos da imagem $2D$ com pontos $3D$ bem conhecidos no mundo, requerendo assim a aquisição de imagens de um objeto de calibração, cuja geometria $3D$ é perfeitamente conhecida (alguns trabalhos nesta área podem ser

verificados em [Tsai, 1987], [Faugeras, 1993], [Heikkilä, 1997], [Batista, 1998], [Zhang, 2000], entre outros);

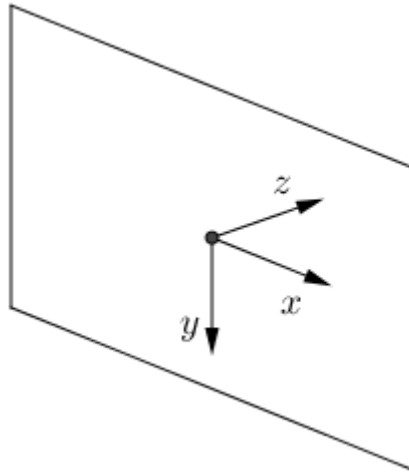
- **Auto-Calibração:** estes métodos associam, entre imagens sucessivas, características da cena ou do objecto a reconstruir, tal como a correspondência de pontos, de tal forma que não requerem a utilização de objectos de calibração (alguns trabalhos nesta área podem ser, por exemplo, encontrados em [Faugeras, 1992], [Pollefeys, 1996], [Triggs, 1997], [Agapito, 1998], [Li, 2004], [Espany, 2007]).

A seguir veremos dois métodos de calibração tradicional.

2.2 Método de Tsai

O método de Tsai na versão coplanar se aplica à casos em que todos os pontos utilizados para calibração estão localizados no mesmo plano. E para utilizá-lo são necessárias algumas considerações, tais como :

- SCM e SCC com mesma orientação. Assim utilizando um referencial de orientação positiva para SCM e \tilde{Z} do SCC apontando para a cena, teremos SCC representado da seguinte maneira.



- Pontos de calibração localizados em $Z = 0$ do SCM.

- Com exceção de f e k_1 , conhecendo os demais parâmetros intrínsecos da câmara podemos determinar (x', y') do plano de projeção a partir de (u, v) . Mas caso não seja conhecido os demais parâmetros e seja conhecido (u_c, v_c) (projeção do centro óptico da câmara). Supondo $\tau = 0$ e $s_x = s_y$ (pixels quadrados, mas se não forem quadrados é necessário conhecer $\frac{s_x}{s_y}$), porém tomando $s_x = s_y = 1$, teremos a seguinte relação:

$$x' = u - u_c \text{ e } y' = v - v_c$$

Observação: com as considerações feitas acima f será dado em unidades de pixels, assim só será possível determinar seu valor real se for conhecido a dimensão de cada pixel.

- É importante escolher uma origem do SCM, de tal forma que sua projeção não seja próxima do centro da imagem ou do eixo x da imagem. Assim podemos garantir que T_y (mais adiante veremos seu significado) não será próximo de 0.

Conhecendo $P_i = (X_i, Y_i, 0)$ (coordenadas dos pontos no SCM) e (u_i, v_i) (correspondentes em pixels), temos:

$$\begin{cases} x'_i = u_i - u_c \\ y'_i = v_i - v_c \\ (x_i, y_i) = (1 + k_1 r^2)(x'_i, y'_i) \end{cases}$$

Onde $r = \sqrt{(x')^2 + (y')^2}$. Assim, sendo a matriz de rotação $R = \begin{bmatrix} r_{xx} & r_{xy} & r_{xz} \\ r_{yx} & r_{yy} & r_{yz} \\ r_{zx} & r_{zy} & r_{zz} \end{bmatrix}$ e o vetor

de translação $T = [T_x \ T_y \ T_z]^t$, teremos a seguinte relação:

$$\begin{aligned} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} &\simeq \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{xx} & r_{xy} & r_{xz} & T_x \\ r_{yx} & r_{yy} & r_{yz} & T_y \\ r_{zx} & r_{zy} & r_{zz} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \\ \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} &\simeq \begin{bmatrix} fr_{xx} & fr_{xy} & fr_{xz} & fT_x \\ fr_{yx} & fr_{yy} & fr_{yz} & fT_y \\ r_{zx} & r_{zy} & r_{zz} & T_z \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \\ \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} &\simeq \begin{bmatrix} fr_{xx}X_i + fr_{xy}Y_i + fr_{xz}Z_i + fT_x \\ fr_{yx}X_i + fr_{yy}Y_i + fr_{yz}Z_i + fT_y \\ r_{zx}X_i + r_{zy}Y_i + r_{zz}Z_i + T_z \end{bmatrix} \end{aligned}$$

Com $Z_i = 0$ e saindo de coordenadas homogêneas temos:

$$\begin{cases} x_i = \frac{fr_{xx}X_i + fr_{xy}Y_i + fr_{xz}Z_i + fT_x}{r_{zx}X_i + r_{zy}Y_i + r_{zz}Z_i + T_z} \\ y_i = \frac{fr_{yx}X_i + fr_{yy}Y_i + fr_{yz}Z_i + fT_y}{r_{zx}X_i + r_{zy}Y_i + r_{zz}Z_i + T_z} \end{cases}$$

Ao dividir x_i por y_i e sabendo que essa razão é a mesma ao dividir x'_i por y'_i , temos:

$$\frac{x'_i}{y'_i} = \frac{fr_{xx}X_i + fr_{xy}Y_i + fr_{xz}Z_i + fT_x}{fr_{yx}X_i + fr_{yy}Y_i + fr_{yz}Z_i + fT_y}$$

$$y'_i X_i r_{xx} + y'_i Y_i r_{xy} + y'_i T_x - x'_i X_i r_{yx} - x'_i Y_i r_{yy} = x'_i T_y$$

$$y'_i X_i \frac{r_{xx}}{T_y} + y'_i Y_i \frac{r_{xy}}{T_y} + y'_i \frac{T_x}{T_y} - x'_i X_i \frac{r_{yx}}{T_y} - x'_i Y_i \frac{r_{yy}}{T_y} = x'_i$$

Sendo $U_1 = \frac{r_{xx}}{T_y}$, $U_2 = \frac{r_{xy}}{T_y}$, $U_3 = \frac{T_x}{T_y}$, $U_4 = \frac{r_{yx}}{T_y}$ e $U_5 = \frac{r_{yy}}{T_y}$, temos:

$$y'_i X_i U_1 + y'_i Y_i U_2 + y'_i U_3 - x'_i X_i U_4 - x'_i Y_i U_5 = x'_i$$

Utilizando uma quantidade n de pontos de calibração, teremos um sistema de equações lineares $AU = b$, onde as linhas a_i de A são $[y'_i X_i \ y'_i Y_i \ y'_i \ -x'_i X_i \ -x'_i Y_i]$ e $b = [x'_1 \ \dots \ x'_n]^t$. O número mínimo necessário de pontos é 5, mas geralmente possuímos uma quantidade maior de pontos, assim será preciso resolver um problema de mínimos quadrados, ou seja, encontrar U tal que $\|AU - b\|$ seja mínimo.

Agora iremos explicar as próximas etapas do método de Tsai:

- Cálculo do módulo de T_y

Sabemos que os elementos r_{xx}, r_{xy}, r_{yx} e r_{yy} podem ser expressos em função de U_i e T_y . $r_{xx} = T_y U_1, r_{xy} = T_y U_2, r_{yx} = T_y U_4$ e $r_{yy} = T_y U_5$. Mas como as linhas de matriz R são unitárias e ortogonais entre si, temos o seguinte sistema:

$$\begin{cases} (U_1^2 + U_2^2)T_y^2 + r_{xz}^2 = 1 \\ (U_4^2 + U_5^2)T_y^2 + r_{yz}^2 = 1 \\ (U_1U_4 + U_2U_5)T_y^2 + r_{xz}r_{yz} = 0 \end{cases}$$

Assim, eliminando r_{xz} e r_{yz} e sendo $S = U_1^2 + U_2^2 + U_4^2 + U_5^2$ e $D = (U_1U_5 - U_2U_4)^2$ temos:

$$T_y^2 = \frac{2}{S \mp \sqrt{S^2 - 4D}}$$

Mas somando as duas primeiras equações do sistema temos que $ST_y \leq 2$, assim:

$$T_y^2 = \frac{2}{S + \sqrt{S^2 - 4D}}$$

- Determinação do sinal de T_y

Repare que o sinal de T_y determina o sinal de outros parâmetros, pois:

$$\begin{aligned} r_{xx} &= U_1T_y \\ r_{xy} &= U_2T_y \\ T_x &= U_3T_y \\ r_{yx} &= U_4T_y \\ r_{yy} &= U_5T_y \end{aligned}$$

Assim inicialmente vamos escolher o sinal positivo para T_y e calcular os demais parâmetros, logo se escolhermos um ponto $P_0 = (X_0, Y_0, 0)$ cuja projeção no plano da imagem (x'_0, y'_0) esteja suficientemente afastada de um dos eixos da imagem. (por exemplo, do eixo y) e verificamos que o sinal de $r_{xx}X_0 + r_{xy}Y_0 + T_x$ é igual ao de x'_0 , mantemos o sinal de T_y caso contrário alteremos o sinal de T_y e dos outros parâmetros que dependem de T_y . Podemos afirmar que essa tentativa funciona, pois o que queremos é um sinal de T_y que permita que o eixo z de câmera esteja orientado na direção da cena, ou seja, o denominador das equações de projeção deve ser positivo para todos os pontos da cena.

- Cálculo dos outros elementos da matriz R

Sabemos que as linhas da matriz R são vetores unitários, assim temos:

$$\begin{aligned} r_{xz} &= \pm \sqrt{1 - r_{xx}^2 - r_{xy}^2} \\ r_{yz} &= \pm \sqrt{1 - r_{yx}^2 - r_{yy}^2} \end{aligned}$$

Considerando inicialmente que r_{xz} é positivo e que r_{yz} será negativo se $r_{xx}r_{yx} + r_{xy}r_{yy} > 0$ e positivo caso contrário. Podemos agora determinar a terceira linha da matriz R , pois:

$$\begin{aligned} r_{zx} &= r_{xy}r_{yz} - r_{xz}r_{yy} \\ r_{zy} &= r_{xz}r_{yx} - r_{xx}r_{yz} \\ r_{zz} &= r_{xx}r_{yy} - r_{xy}r_{yx} \end{aligned}$$

- Cálculo aproximado de f e T_z

Considerando $k_1 = 0$, assim $(x_i, y_i) = (x'_i, y'_i)$. Temos as seguintes equações que dependem de f e T_z :

$$\begin{aligned} (r_{xx}X_i + r_{xy}Y_i + T_x)f - x'_iT_z &= x'_i(r_{zx}X_i + r_{zy}Y_i) \\ (r_{yx}X_i + r_{yy}Y_i + T_y)f - y'_iT_z &= y'_i(r_{zx}X_i + r_{zy}Y_i) \end{aligned}$$

Onde obtemos os valores de f e T_z resolvendo um problema de mínimos quadrados.

Repare que, caso encontremos um valor negativo para f , isso significa que o sinal atribuído inicialmente para r_{zx} estava incorreto e assim devemos trocá-lo, assim como os sinais de r_{yz} , r_{zx} , r_{zy} , f e T_z .

- Cálculo exato de f , T_z e k_1 .

Se quisermos encontrar o valor de k_1 que foi considerado como zero na etapa anterior, devemos fazer a seguinte alteração nas equações que foram usadas para encontrar f e T_z :

$$\begin{aligned}(r_{xx}X_i + r_{xy}Y_i + T_x)f - \frac{x'_i(r_{zx}X_i + r_{zy}Y_i + T_z)}{1+k_1((x'_i)^2 + (y'_i)^2)} &= 0 \\ (r_{yx}X_i + r_{yy}Y_i + T_y)f - \frac{y'_i(r_{zx}X_i + r_{zy}Y_i + T_z)}{1+k_1((x'_i)^2 + (y'_i)^2)} &= 0\end{aligned}$$

Que são não-lineares. Portanto temos um problema de regressão não-linear cujo objetivo é minimizar a soma dos quadrados dos erros nas n equações assim obtidas.

2.2.1 Implementação do Método de Tsai

Na implementação do Método de Tsai abaixo foi desconsiderado a distorção radial (ou seja, k_1). Os valores **m** e **n** passados representam o tamanho da imagem em pixel, ou seja, se temos uma imagem 640×480 , m será igual a 640 e n será igual a 480 colunas. Já a matriz **pontos de calibração** terá 5 colunas (1ª coluna: coordenadas u ; 2ª coluna: coordenadas v ; 3ª coluna: coordenadas X ; 4ª coluna: coordenadas Y ; 5ª coluna: coordenadas $Z = 0$) e o número de linhas depende da quantidade de pontos de calibração que temos disponível. Assim o código irá retornar o valor de f (distância focal da câmera expressa em pixels), a matriz R (matriz de rotação do SCM para SCC), o vetor T (vetor de translação do SCM para SCC), o posicionamento da câmera (que é calculado convertendo a origem do SCC para o SCM, através da equação $P = R^t\tilde{P} - R^tT$, ou seja, basta calcular $-R^tT$) e o ângulo de visão da câmera (sabendo que a tangente do arco metade de este ângulo é calculada a partir da razão $\frac{v_c}{f}$).

```
function [f,camera,T,R,angulo] = parametros(m,n,pontoscalibracao)
[a,b]=size(pontoscalibracao)
aux=ones(a,2)
aux(:,1)=aux(:,1)*(m/2)
aux(:,2)=aux(:,2)*(n/2)
pontos=pontoscalibracao
pontos(:,1)=pontos(:,1)-aux(:,1)//determinando x'
pontos(:,2)=pontos(:,2)-aux(:,2)//determinando y'

//Determinando a matriz A, tal que AU = b

A=zeros(a,b)
A(:,1)=pontos(:,2).*pontos(:,3)
A(:,2)=pontos(:,2).*pontos(:,4)
A(:,3)=pontos(:,2)
A(:,4)=-pontos(:,1).*pontos(:,3)
A(:,5)=-pontos(:,1).*pontos(:,4)
```

```

//Achar o valor de U tal que ||AU-b|| seja minimo

SolU=lsq(A,pontos(:,1))

//Calculo de Ty

S=sum(SolU.*SolU)-(SolU(3))^2
D=(SolU(1)*SolU(5)-SolU(2)*SolU(4))^2
T=[0 0 0]';
T(2)=sqrt(2/(S+sqrt(S^2-4*D)))

//Determinacao do sinal de Ty

R=eye(3,3)
R(1,1)=SolU(1)*T(2)
R(1,2)=SolU(2)*T(2)
T(1)=SolU(3)*T(2)
R(2,1)=SolU(4)*T(2)
R(2,2)=SolU(5)*T(2)
if (R(1,1)*pontos(1,3)+R(1,2)*pontos(1,4)+T(1))*pontos(1,1)<0
then
    T(2)=-T(2)
    R(1,1)=SolU(1)*T(2)
    R(1,2)=SolU(2)*T(2)
    T(1)=SolU(3)*T(2)
    R(2,1)=SolU(4)*T(2)
    R(2,2)=SolU(5)*T(2)
end

//Calculo dos demais elementos da matriz de rotacao R

R(1,3)=sqrt(1-(R(1,1))^2-(R(1,2))^2)
if R(1,1)*R(2,1)+R(1,2)*R(2,2)>0 then
R(2,3)=-sqrt(1-(R(2,1))^2-(R(2,2))^2)
else
R(2,3)=sqrt(1-(R(2,1))^2-(R(2,2))^2)
end
//Para determinar a terceira linha de R
R(3,:)=cross(R(1,:),R(2,:))

//Calculo aproximado de f e Tz

B=[R(1,1)*pontos(:,3)+R(1,2)*pontos(:,4)+ones(a,1)*T(1),
-pontos(:,1); R(2,1)*pontos(:,3)+R(2,2)*pontos(:,4)+
ones(a,1)*T(2),-pontos(:,2)]
b=[pontos(:,1).*(R(3,1)*pontos(:,3)+R(3,2)*pontos(:,4));
pontos(:,2).*(R(3,1)*pontos(:,3)+R(3,2)*pontos(:,4))]
aux2=lsq(B,b)
f=aux2(1)
T(3)=aux2(2)

```

```

//Consertando o sinal de rxz

if f<0 then
    R(3, 1:2) = -R(3, 1:2);
    R(1:2, 3) = -R(1:2, 3);
    f=-f
    T(3)=-T(3)
end

//Achando a posicao da camera

camera=-R'*T

//Achando o angulo de visao da camera

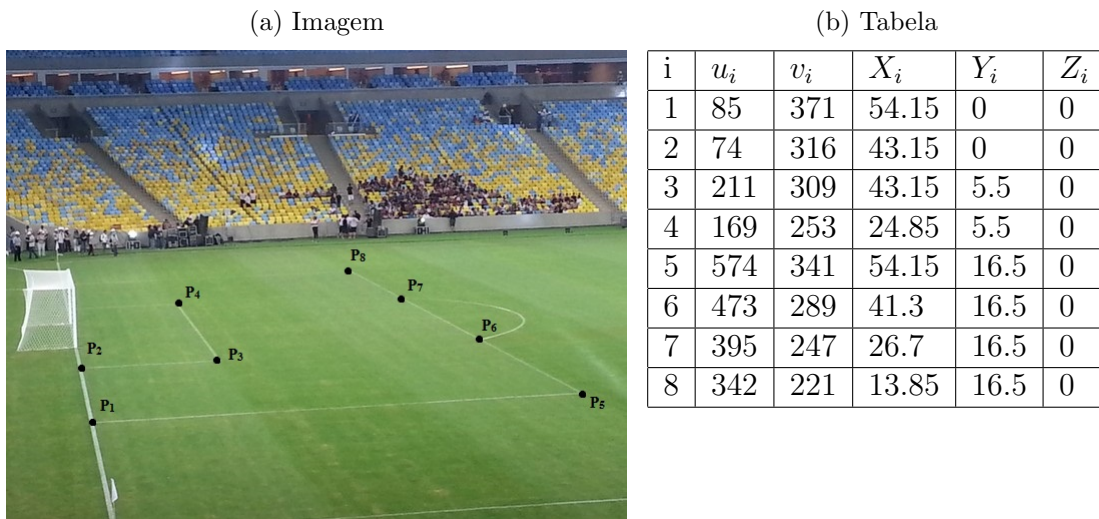
angulo = 2*atand((n/2)/f)
endfunction

```

2.2.2 Exemplos de Aplicação do Método de Tsai

• **Primeiro Exemplo:** Foi utilizado uma foto tirada do estádio do Maracanã, onde o SCM foi fixado com origem na bandeira de escanteio, com eixo X na direção da linha de fundo, eixo Y na direção da linha lateral e eixo Z para cima. O SCI tem origem no centro da imagem, eixo horizontal orientado para a direita e eixo vertical orientado para baixo. Segue a imagem (620x470) com os pontos de calibração marcados (pode ser calibrada usando o método de Tsai para pontos coplanares devido ao fato de que, na fotografia, aparece um número suficiente de pontos de coordenadas conhecidas no espaço) e uma tabela que representa a matriz pontos de calibração.

Figura 1: Imagem e Tabela do Exemplo 1 de Tsai

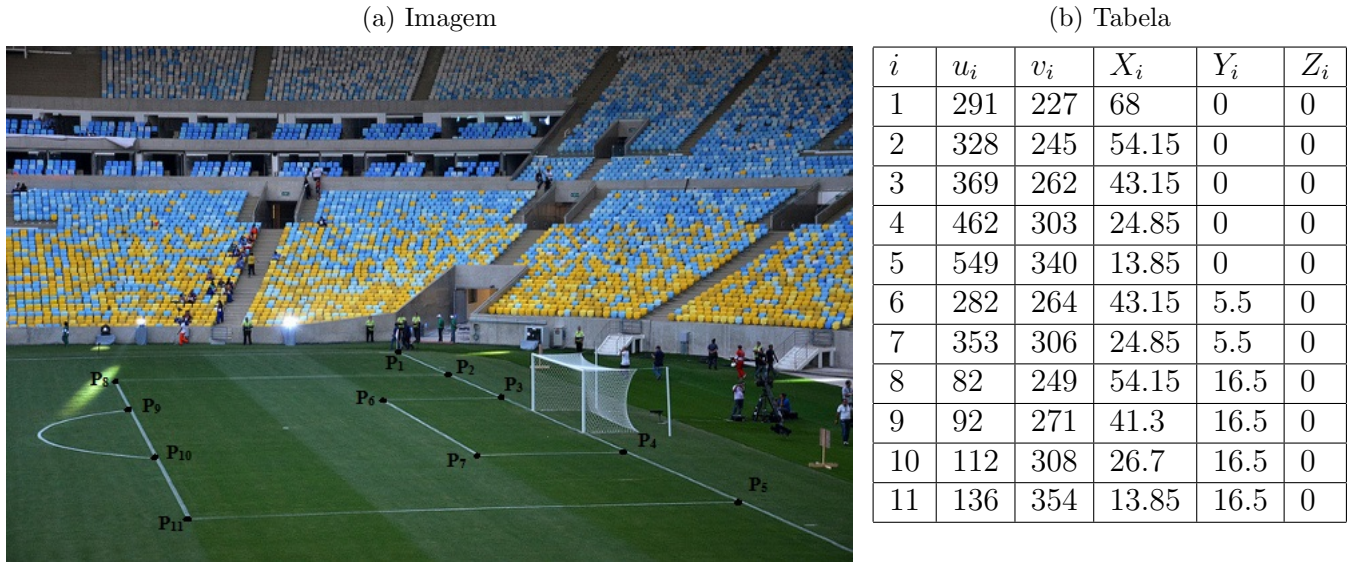


Para esta imagem obtemos um posicionamento para a câmera $[107.85314; -2.2320406; 10.466203]$,

além de $R = \begin{bmatrix} 0,1819 & 0,9829 & -0,0282 \\ 0,1096 & -0,0487 & -0,9928 \\ -0,9772 & 0,1775 & -0,1166 \end{bmatrix}$ e $T = [-17.128097 \quad -1.5424675 \quad 107.00969]^t$

• **Segundo Exemplo:** Seguindo os mesmos critérios do primeiro exemplo, segue a imagem (634x387) que também foi tirada do Maracanã, porém de outro ponto e uma tabela que representa a matriz pontos de calibração.

Figura 2: Imagem e Tabela do Exemplo 2 de Tsai



Para esta imagem obtemos um posicionamento para a câmera $[-38.769718; 21.246361; 9.0540792]$, além de $R = \begin{bmatrix} -0.2130157 & -0.9769634 & 0.0129154 \\ -0.0585364 & -0.0004341 & -0.9982852 \\ 0.9752937 & -0.2134064 & -0.0570954 \end{bmatrix}$ e $T = [12.381422 \quad 6.7783362 \quad 42.862919]^t$

• **Terceiro Exemplo:** Para este exemplo foi tirada uma foto de um quadro, onde as coordenadas do mundo dos pontos utilizados para a calibração foram obtidas manualmente com o auxílio de instrumentos de medida (fita métrica, régua e esquadros), com o eixo x no canto esquerdo da imagem pintada (sentido positivo para baixo), eixo y no canto superior (sentido positivo para a direita) e eixo z com sentido positivo para fora. Assim, sabendo que a imagem obtida possui tamanho 3264x1836, pode-se determinar as coordenadas em pixels dos respectivos pontos marcados. Abaixo segue a imagem com a tabela dos pontos.

Para esta imagem obtemos um posicionamento para a câmera $[1.4145454; 64.3478; 113.16571]$, além de $R = \begin{bmatrix} -0.0059214 & -0.9677899 & -0.2516896 \\ 0.9778996 & 0.058207 & 0.2008094 \\ 0.2089913 & -0.2449381 & -0.9467460 \end{bmatrix}$ e $T = [-33.78418 \quad -27.853508 \quad 122.60479]^t$

2.3 Método de Zhang

É um método que é uma alternativa para padrões tridimensionais, mas que usa um padrão bidimensional posicionado em diversas posições no espaço, gerando várias imagens. Onde cada uma destas imagens possuem n pontos conhecidos no sistema de referência padrão. Além disso, os parâmetros intrínsecos são os mesmos para todas as imagens, mas os extrínsecos não (alte-

Figura 3: Imagem e Tabela do Exemplo 3 de Tsai

(a) Imagem



(b) Tabela

i	u_i	v_i	X_i	Y_i	Z_i
1	880	434	13.6	12.7	0
2	1228	228	7.1	23.3	0
3	1612	164	5.3	34.2	0
4	1307	708	20.6	25.4	0
5	1405	705	20.6	28.3	0
6	1528	708	20.6	31.9	0
7	1626	707	20.6	34.8	0
8	1859	244	6.6	41.5	0
9	1949	241	6.6	44.2	0
10	1864	640	18.3	41.5	0

ram quando o padrão é alterado de posição).

Como temos mais de uma imagem, podemos ajustar o modelo da câmera de forma mais geral. Observe a transformação em coordenadas homogêneas abaixo:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \simeq \begin{bmatrix} \alpha & c & u_c \\ 0 & \beta & v_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & T \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Restringindo esta transformação ao plano $Z = 0$ será definida uma transformação projetiva (ou homografia) H entre este plano e o plano da imagem, cuja expressão em coordenadas homogêneas é dada por:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \simeq \begin{bmatrix} \alpha & c & u_c \\ 0 & \beta & v_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_1 & r_2 & T \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

Com r_1 e r_2 sendo as duas primeiras colunas da matriz R .

- **Estimação das homografias:** Chamamos a transformação linear que associa as coordenadas do ponto do mundo às suas projeções na imagem de homografia, denotada por H , assim para cada imagem teríamos uma homografia que é dada pela expressão abaixo:

$$s \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} \simeq H_i \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix}$$

Assim temos que $H_i = \lambda K \begin{bmatrix} r_1 & r_2 & T \end{bmatrix}$.

Para determinar a homografia que transforma as coordenadas do mundo em coordenadas da imagem, temos em coordenadas homogêneas que:

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix}$$

Assim:

$$\begin{cases} u_i = \frac{H_{11}X_i + H_{12}Y_i + H_{13}}{H_{31}X_i + H_{32}Y_i + H_{33}} \\ v_i = \frac{H_{21}X_i + H_{22}Y_i + H_{23}}{H_{31}X_i + H_{32}Y_i + H_{33}} \end{cases}$$

$$\begin{cases} u_i H_{31}X_i + u_i H_{32}Y_i + u_i H_{33} - H_{11}X_i - H_{12}Y_i - H_{13} = 0 \\ v_i H_{31}X_i + v_i H_{32}Y_i + v_i H_{33} - H_{21}X_i - H_{22}Y_i - H_{23} = 0 \end{cases}$$

Escrevendo na forma matricial temos:

$$\begin{bmatrix} -X_i & -Y_i & -1 & 0 & 0 & 0 & u_i X_i & u_i Y_i & u_i \\ 0 & 0 & 0 & -X_i & -Y_i & -1 & v_i X_i & v_i Y_i & v_i \end{bmatrix} \begin{bmatrix} H_{11} \\ H_{12} \\ H_{13} \\ H_{21} \\ H_{22} \\ H_{23} \\ H_{31} \\ H_{32} \\ H_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Sendo assim, se a imagem possui um total de n pontos, teremos $2n$ equações e portanto a matriz da esquerda é $2n \times 9$ e o vetor de zeros terá $2n$ linhas. Para encontrar o valor de h que está escrito como $Mh = 0$ podemos resolver por decomposição de valores singulares da matriz M , ou seja, devemos escrever $M = U.S.V^T$, onde U é $2n \times 2n$, S é $2n \times 9$ e V é 9×9 , os elementos da diagonal de S são chamados de valores singulares da decomposição de M , e cada valor dessa diagonal está associado a uma coluna de U (vetor singular a esquerda) e a uma linha de V^T , ou seja, coluna de V (vetor singular a direita), assim a solução h será dada pelo vetor v_k (coluna) da matriz V que está associado ao menor valor singular da diagonal de S . Porém antes de achar o valor de h é sugerido normalizar a matriz que contém as coordenadas do mundo e da imagem.

Sendo N_u e N_x (matrizes 3×3) a normalização das coordenadas da imagem e do mundo respectivamente temos:

$$\begin{aligned} N_u u_i &= H' N_x X_i \\ N_u H X_i &= H' N_x X_i \\ N_u H &= H' N_x \\ H &= N_u^{-1} H' N_x \end{aligned}$$

Por fim dado uma solução inicial para minimizar o erro de projeção devemos utilizar de uma otimização, que geralmente é implementada com esquemas iterativos, como o método de Levenberg-Marquart, ou seja, minimizar:

$$\sum_{i=1}^n \|\widehat{U}_i - H\widehat{X}_i\|^2$$

Onde \widehat{U}_i é o mesmo que $(u_i, v_i)^T$ (ponto de coordenadas da imagem) e \widehat{X}_i é o mesmo que $(X_i, Y_i, 1)^T$ (ponto de coordenadas do mundo escrito como coordenadas homogêneas)

- Estimação dos parâmetros intrínsecos e extrínsecos.

Neste passo as Homografias já estarão estimadas para cada imagem, agora podemos determinar os parâmetros intrínsecos (o mesmo para todas as imagens) e extrínseco (que será diferente para cada uma das imagens). Lembrando que $H = \lambda K \begin{bmatrix} r_1 & r_2 & T \end{bmatrix}$, onde λ é diferente de zero. Assim sendo h_1 , h_2 e h_3 as colunas de H , temos:

$$\begin{aligned}h_1 &= \lambda K r_1 \Leftrightarrow K^{-1} h_1 = \lambda r_1 \\h_2 &= \lambda K r_2 \Leftrightarrow K^{-1} h_2 = \lambda r_2\end{aligned}$$

Com isso temos que:

$$\begin{aligned}h_1^T (K^{-1})^T &= \lambda r_1^T \\h_2^T (K^{-1})^T &= \lambda r_2^T\end{aligned}$$

Como $r_1^T r_2 = r_2^T r_1 = 0$ e $r_1^T r_1 = r_2^T r_2 = 1$, podemos escrever as equações acima da seguinte forma:

$$\begin{aligned}h_1^T (K^{-1})^T K^{-1} h_2 &= 0 \\h_1^T (K^{-1})^T K^{-1} h_1 &= h_2^T (K^{-1})^T K^{-1} h_2\end{aligned}$$

Sendo $B = (K^{-1})^T K^{-1}$, obtemos:

$$\begin{cases} B_{11} = \frac{1}{\alpha^2} \\ B_{12} = B_{21} = -\frac{c}{\alpha^2 \beta} \\ B_{13} = B_{31} = \frac{v_c c - u_c \beta}{\alpha^2 \beta} \\ B_{22} = \frac{c^2}{\alpha^2 \beta^2} + \frac{1}{\beta^2} \\ B_{23} = B_{32} = \frac{c(v_c c - u_c \beta)}{\alpha^2 \beta^2} - \frac{v_c}{\beta^2} \\ B_{33} = \frac{(v_c c - u_c \beta)^2}{\alpha^2 \beta^2} + \frac{v_c^2}{\beta^2} + 1 \end{cases}$$

Se encontrarmos os termos da matriz B , podemos determinar os parâmetros intrínsecos, logo seguiremos os seguintes passos:

$$\begin{cases} h_1^T B h_2 = 0 \\ h_1^T B h_1 - h_2^T B h_2 = 0 \end{cases}$$

Com $h_p = (H_{p1}, H_{p2}, H_{p3})^T$, $b = (B_{11}, B_{12}, B_{22}, B_{13}, B_{23}, B_{33})^T$ e $v_{pq}(H) = (H_{1p}H_{1q}, H_{1p}H_{2q} + H_{2p}H_{1q}, H_{2p}H_{2q}, H_{3p}H_{1q} + H_{1p}H_{3q}, H_{3p}H_{2q} + H_{2p}H_{3q}, H_{3p}H_{3q})$ temos:

$$h_p^T B h_q = v_{pq}(H) b$$

Escrevendo na forma matricial:

$$\begin{bmatrix} v_{12}(H) \\ v_{11}(H) - v_{22}(H) \end{bmatrix} b = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Repare que ao escrever na forma de matriz, a matriz do lado esquerdo possuirá 2 equações para cada M imagens, portanto $2M$ equações, sendo assim a matriz será $2M \times 6$ e o vetor de zeros possuirá $2M$ linhas. Para encontrar o vetor b , também podemos usar decomposição em valores singulares, assim como na determinação da homografia. Com isso obtemos os seguintes parâmetros intrínsecos:

$$\begin{cases} \alpha = \sqrt{w/(dB_{11})} \\ \beta = \sqrt{w/d^2 B_{11}} \\ c = \sqrt{w/(d^2 B_{11})} B_{12} \\ u_c = (B_{12}B_{23} - B_{22}B_{13})/d \\ v_c = (B_{12}B_{13} - B_{11}B_{23})/d \end{cases}$$

com

$$\begin{cases} w = B_{11}B_{22}B_{33} - B_{12}^2B_{33} - B_{11}B_{23}^2 + 2B_{12}B_{13}B_{23} - B_{22}B_{13}^2 \\ d = B_{11}B_{22} - B_{12}^2 \end{cases}$$

E para determinar os parâmetros extrínsecos de cada imagem, sabendo que cada imagem possui uma Homografia temos:

$$\begin{cases} r_1 = \lambda K^{-1}h_1 \\ r_2 = \lambda K^{-1}h_2 \\ T = \lambda K^{-1}h_3 \\ r_3 = r_1 \times r_2 \end{cases}$$

com

$$\lambda = \frac{1}{\|K^{-1}h_1\|} = \frac{1}{\|K^{-1}h_2\|}$$

• Estimação dos coeficientes de distorção: O método de Zhang considera os coeficientes k_1 e k_2 de distorção radial, definidos pela relação entre pontos com distorção (x, y) e sem distorção (x', y') da imagem:

$$\begin{cases} x = x' + x'(k_1r^2 + k_2r^4) \\ y = y' + y'(k_1r^2 + k_2r^4) \\ r = \sqrt{x'^2 + y'^2} \end{cases}$$

Onde (x, y) se relacionada com os pontos observados (u, v) da imagem por:

$$\begin{cases} u = \alpha x + cy + u_c \\ v = \beta y + v_c \end{cases}$$

Logo temos:

$$\begin{cases} u = u' + (u' - u_c)(k_1r^2 + k_2r^4) \\ v = v' + (v' - v_c)(k_1r^2 + k_2r^4) \end{cases}$$

Na forma matricial:

$$\begin{bmatrix} (u' - u_c)r^2 & (u' - u_c)r^4 \\ (v' - v_c)r^2 & (v' - v_c)r^4 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} u - u' \\ v - v' \end{bmatrix}$$

Se temos n pontos e m imagens, a matriz do lado esquerdo possui $2nm$ linhas, assim como o vetor do lado direito. Para determinar os valores k_1 e k_2 podemos resolver um problema de mínimos quadrados.

2.3.1 Implementação do Método de Zhang

• **Primeira Etapa** (Estimação da Homografia): Neste etapa, para cada imagem, será determinada uma homografia. Para isso na função *Homografia* é passado uma matriz A , onde a primeira coluna é composta pelas coordenadas u da imagem, a segunda pelas coordenadas v , a terceira pelas coordenadas X do mundo e a quarta pelas coordenadas Y . Como retorno desta função temos a homografia, que é calculada usando a normalização das coordenadas do mundo e da imagem a partir da função *normalizar*, que recebe a mesma matriz A já mencionada e retorna a normalização. Com a homografia inicial determinada é necessário otimizá-la e para isso foi criada a função *LM* que utiliza a função já definida pelo Scilab, *lsqr solve* (otimização não linear Levenberg-Marquardt), observe os passos utilizados para modelar nosso problema para função já existente no Scilab.

Como a função *lsqr solve* tem como entrada um vetor, transformamos a matriz H que é 3×3 em um vetor de 9 linhas. Assim a função que queremos otimizar $b - ax$ é tal que o x inicial é a homografia inicial e a e b para um único ponto da imagem é dado por:

$$a = \begin{bmatrix} X_i & Y_i & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & X_i & Y_i & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & X_i & Y_i & 1 \end{bmatrix} \text{ e } b = \begin{bmatrix} u_i \\ v_i \end{bmatrix}$$

Se temos n pontos, a será $3n \times 9$ e b será $2n \times 1$. Repare que antes de fazer a subtração $b - ax$ o termo ax é transformado, ou seja deixamos de ter coordenadas homogêneas.

```
function H = Homografia(A)
    [m,n] = size(A);
    X = zeros(2,m);
    U = zeros(2,m);
    X(1,:) = A(:,3)';
    X(2,:) = A(:,4)';
    U(1,:) = A(:,1)';
    U(2,:) = A(:,2)';
    NX = normalizar(X);
    NU = normalizar(U);
    Xhomg = [X(1,:); X(2,:); ones(1,m)];
    Uhomg = [U(1,:); U(2,:); ones(1,m)];
    X = NX*Xhomg;
    U = NU*Uhomg;
    M = zeros(2*m,9)
    for i=1:m
        M(2*i-1,:) = [-X(1,i),-X(2,i),-1,0,0,0,U(1,i)*
            X(1,i),U(1,i)*X(2,i),U(1,i)];
        M(2*i,:) = [0,0,0,-X(1,i),-X(2,i),-1,U(2,i)*
            X(1,i),U(2,i)*X(2,i),U(2,i)];
    end
    [U,S,V] = svd(M,0);
    h = V(:,9);
    Hnorm = [h(1),h(2),h(3); h(4),h(5),h(6);h(7),h(8),h(9)];
    H = inv(NU)*Hnorm*NX
```

endfunction

```
function N = normalizar(X)
    [m,n] = size(X);
    MatrizSoma = sum(X,2);
    xbarra = (1/n)*MatrizSoma(1);
    ybarra = (1/n)*MatrizSoma(2);
    aux1 = 0;
    aux2 = 0;
    for i=1:n
        sigmax = (1/n)*(X(1,i)-xbarra)^2;
        aux1 = aux1+sigmax;
        sigmay = (1/n)*(X(2,i)-ybarra)^2;
        aux2 = aux2+sigmay;
    end
    sx = sqrt(2/aux1);
    sy = sqrt(2/aux2);
    N = [sx 0 -sx*xbarra; 0 sy -sy*ybarra; 0 0 1];
```

endfunction

```
function [a,b,Hvetor]= Termos(H,A)
    [n,m] = size(A);
    Hvetor = [H(1,1);H(1,2);H(1,3);H(2,1);H(2,2);
    H(2,3);H(3,1);H(3,2);H(3,3)];
    a = zeros(3*n,9);
    b = zeros(2*n,1);
    for i=1:n
        a(3*i-2,:) = [A(i,3),A(i,4),1,0,0,0,0,0,0];
        a(3*i-1,:) = [0,0,0,A(i,3),A(i,4),1,0,0,0];
        a(3*i,:) = [0,0,0,0,0,0,A(i,3),A(i,4),1];
        b(2*i-1,1) = A(i,1);
        b(2*i,1) = A(i,2);
    end
```

endfunction

```
function B = modifica(A)
    [m,n] = size(A);
    m = m/3;
    k = 2*m
    B = zeros(k,1);
    for i=1:m
        B(2*i-1,1) = A(3*i-2,1)/A(3*i,1);
        B(2*i,1) = A(3*i-1,1)/A(3*i,1);
    end
```

endfunction

```
function y = f(x,m)
    B = a*x;
```

```

    y = b - modifica(B);
endfunction

function Hnovo = LM(H,A)
    [m,n] = size(A);
    [a,b,Hvetor]= Termos(H,A);
    [H1,v] = lsqrsolve(Hvetor,f,2*m);
    Hnovo = [H1(1),H1(2),H1(3);H1(4),H1(5),H1(6);H1(7),H1(8),H1(9)];
endfunction

```

• **Segunda Etapa** (Estimação dos Parâmetros da Câmera): Na função *Intriseccos* a entrada é uma matriz composta por todas as homografias já encontradas no passo anterior e o retorno é matriz com os parâmetros intrínsecos da câmera. Já na *Extrinsecos* a entrada é a matriz dos parâmetros intrínsecos e a homografia de cada imagem e o retorno é a matriz R (matriz de rotação do SCM para SCC), o vetor T (vetor de translação do SCM para SCC), o posicionamento da câmera (que é calculado convertendo a origem do SCC para o SCM, através da equação $P = R^t \tilde{P} - R^t T$, ou seja, basta calcular $-R^t T$).

```

function K = Intrinsecos(Hconjunto)
    [a,m] = size(Hconjunto)
    m = m/3
    V = zeros(2*m,6)
    for n = 1:m
        H = [Hconjunto(:,3*n-2) Hconjunto(:,3*n-1) Hconjunto(:,3*n)];
        v11 = [H(1,1)*H(1,1), H(1,1)*H(2,1)+H(2,1)*H(1,1), H(2,1)*
            H(2,1), H(3,1)*H(1,1)+H(1,1)*H(3,1), H(3,1)*H(2,1)+H(2,1)*
            H(3,1), H(3,1)*H(3,1)];
        v12 = [H(1,1)*H(1,2), H(1,1)*H(2,2)+H(2,1)*H(1,2), H(2,1)*
            H(2,2), H(3,1)*H(1,2)+H(1,1)*H(3,2), H(3,1)*H(2,2)+H(2,1)*
            H(3,2), H(3,1)*H(3,2)];
        v22 = [H(1,2)*H(1,2), H(1,2)*H(2,2)+H(2,2)*H(1,2), H(2,2)*
            H(2,2), H(3,2)*H(1,2)+H(1,2)*H(3,2), H(3,2)*H(2,2)+H(2,2)*
            H(3,2), H(3,2)*H(3,2)];
        V(n*2-1,:) = v12;
        V(n*2,:) = (v11-v22);
    end
    [U,S,V1] = svd(V,0);
    b = V1(:,6);
    w = b(1)*b(3)*b(6) - b(2)*b(2)*b(6)-b(1)*b(5)*
        b(5)+2*b(2)*b(4)*b(5)-b(3)*b(4)*b(4);
    d = b(1)*b(3)-b(2)*b(2);
    alpha = sqrt(w/(d*b(1)));
    betaa = sqrt(w/(d*d) * b(1));
    c = sqrt(w/(d*d* b(1)))*b(2);
    uc = (b(2)*b(5) - b(3)*b(4))/d;
    vc = (b(2)*b(4)-b(1)*b(5))/d;
    K = [alpha,c,uc;0, betaa, vc;0, 0, 1];
endfunction

```

```

function [R,T,camera] = Extrinsecos(K,H)
    Kinv = inv(K);
    lambda1 = 1/(norm(Kinv *H(:,1)));
    lambda2 = 1/(norm(Kinv *H(:,2)));
    lambda = (lambda1 + lambda2)/2;
    r1 = lambda*Kinv*H(:,1);
    r2 = lambda*Kinv*H(:,2);
    r3 = cross(r1,r2);
    T = lambda*Kinv*H(:,3);
    R = [r1,r2,r3];
    camera=-R'*T
endfunction

```

• **Terceira Etapa** (Estimação da Distorção da Lente): Nesta etapa a função *Distorcao* recebe a matriz dos parâmetros intrínsecos, uma matriz W que é composta por todas as matrizes $R's$ e vetores $T's$ e a matriz A que contém as coordenadas do mundo e da imagem. Como retorno temos os valores k_1 e k_2 de distorção da lente.

```

function k = Distorcao(K,W,A)
    [n,a] = size(A);
    [b,m] = size(W);
    m = m/4;
    uc = K(1,3);
    vc = K(2,3);
    D = zeros(2*m*n,2);
    d = zeros(2*m*n,1);
    l = 1
    for i = 1:m
        for j = 1:n
            M1=[W(:,4*i-3) W(:,4*i-2) W(:,4*i-1) W(:,4*i)]
                *[A(j,4*m-1);A(j,4*m);0;1];
            x = M1(1)/M1(3);
            y = M1(2)/M1(3);
            r = sqrt(x^2 + y^2);
            M2 = [K(1,:);K(2,:)]*[x;y;1];
            u = M2(1);
            v = M2(2);
            du = u-uc;
            dv = v-vc;
            D(2*l-1,:)= [du*r^2,du*r^4];
            D(2*l,:) = [dv*r^2,dv*r^4];
            up = A(j,4*m-3);
            vp = A(j,4*m-2);
            d(2*l-1,:)=up - u;
            d(2*l,:)=vp - v;
            l = l+1
        end
    end

```

```

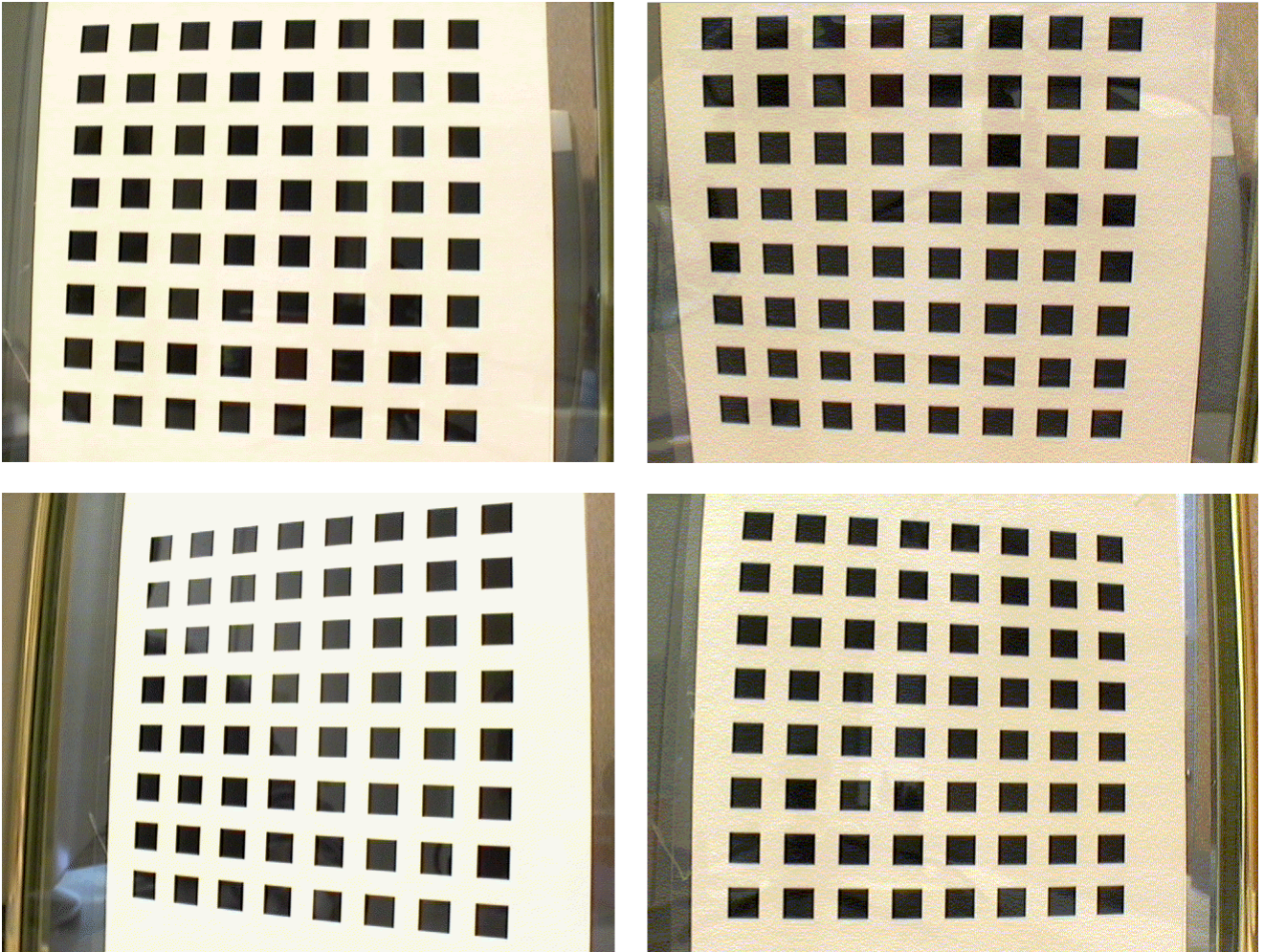
end
k = lsq(D,d)
endfunction

```

2.3.2 Exemplo de Aplicação do Método de Zhang

• **Primeiro Exemplo** Para este método o exemplo utilizado é o tradicional (folha composta por quadrados). Onde temos 4 imagens que foram tiradas de pontos diferentes. Com origem das coordenadas do mundo localizada na extremidade inferior esquerda (vértice do primeiro quadrado preto), eixo x com sentido positivo para a direita, eixo y com sentido positivo para baixo e eixo z com sentido positivo "furando" a folha. Segue as imagens com a tabela dos pontos utilizados.

Figura 4: Figura do Exemplo do Método de Zhang



Sendo assim, seguindo o mesmo objetivo do método de Tsai, o valor obtido para o posicionamento das câmeras foram: primeira câmera = $[6.78; -2.42; -14.76]$; segunda câmera = $[5.39; -6.85; -14.48]$; terceira câmera = $[10.38; -2.49; -14.31]$ e quarta câmera = $[1.26; -2.62; -16.00]$.

Com as respectivas matrizes de rotação e vetor de translação: $R_1 = \begin{bmatrix} 0.9868437 & -0.0262181 & 0.1595370 \\ -0.0192706 & 0.9606552 & 0.2770744 \\ -0.1605244 & -0.2765035 & 0.9475113 \end{bmatrix}$

Tabela 1: Tabela do Exemplo do Método de Zhang

i	X_i	Y_i	u_i	v_i	u_i	v_i	u_i	v_i	u_i	v_i
1	0	-0.5	63	405	74	409	137	394	84	410
2	0.89	-0.5	116	409	127	412	180	399	141	411
3	1.78	-0.5	170	412	181	414	225	404	198	411
4	2.67	-0.5	226	415	237	417	274	409	255	411
5	3.56	-0.5	284	418	293	419	325	414	310	411
6	0	0	62	436	77	436	136	421	83	441
7	0.89	0	115	440	129	439	179	427	141	442
8	1.78	0	169	444	183	443	224	433	198	442
9	2.67	0	226	447	237	445	272	439	254	442
10	3.56	0	283	450	293	448	323	445	310	442

$$\begin{aligned}
e \ T_1 &= [-4.3577936 \ 6.5508253 \ 14.403652]^T; R_2 = \begin{bmatrix} 0.9974326 & -0.0147460 & 0.0773746 \\ 0.0039787 & 0.9994510 & 0.0000448 \\ -0.0774127 & 0.0010996 & 0.9969436 \end{bmatrix} e \\
T_2 &= [-4.2852213 \ 6.7746415 \ 14.865609]^T; R_3 = \begin{bmatrix} 0.9216913 & -0.0567913 & 0.4109205 \\ -0.0819790 & 0.9489556 & 0.2723353 \\ -0.4096791 & -0.2702305 & 0.8699884 \end{bmatrix} e \\
T_3 &= [-3.5526597 \ 6.9639301 \ 15.952984]^T; R_4 = \begin{bmatrix} 0.9853992 & -0.0197745 & -0.1691078 \\ 0.0636121 & 0.9640649 & 0.2579385 \\ 0.1579302 & -0.2649297 & 0.9512467 \end{bmatrix} e \\
T_4 &= [-4.0020293 \ 6.5745642 \ 14.327396]^T
\end{aligned}$$

3 Conclusão

Introduzimos este trabalho com o conceito de geometria projetiva, que permite obtermos uma correspondência entre os pares de pontos $2D$ e $3D$. A partir deste ponto, o objetivo foi relacionar este ramo da geometria com conceitos de visão computacional, principalmente na área ligada à calibração de câmeras, pois como sabemos, a calibração consiste no processo de determinar dados geométricos digitais e ópticos da câmera, admitindo que sejam conhecidos um conjunto de pares de pontos bidimensionais em uma imagem e seus respectivos pontos tridimensionais e com isso, fica evidente a necessidade de definir relações entre as coordenadas de pontos $3D$ com as coordenadas $2D$ de imagens dos mesmos, o que é feito com o auxílio da geometria projetiva.

Relacionando o conceitos de geometria projetiva e calibração podemos definir dois métodos muito conhecidos (Tsai e Zhang), ambos com metodologias diferentes. O primeiro necessitando de uma única imagem para determinar parâmetros intrínsecos e extrínsecos da câmera e o segundo necessitando de mais de uma imagem. Com isso, dado alguns exemplos, conseguimos alcançar nosso objetivo principal, que era de determinar o posicionamento das câmeras utilizadas para tirar as respectivas fotos.

A partir dos resultados alcançados, podemos concluir que, esse trabalho é útil em ramos como a robótica, onde podemos ajudar um robô a "enxergar" e portanto se movimentar. Além disso, pensando nos exemplos em foi tirado uma foto de um campo de futebol, podemos utilizar os dados obtidos no experimento em tira teimas de futebol (sistema que flagra jogadores de futebol em impedimento, por exemplo), porém o sistema é suscetível à falha humana, pois são operadores que indicam, manualmente, a posição dos jogadores na tela.

4 Referências Bibliográficas

H. S. M. Coxeter, **Introduction to Geometry** (2nd ed.), Wiley, 1969, New York.

H. S. M. Coxeter, **Projective Geometry** (2nd ed.), Univ. of Toronto Press, 1974, Toronto.

Paulo Cezar Carvalho e Luiz Velho, **Fotografia 3D**, 25º Colóquio Brasileiro de Matemática, IMPA.

Luciano de Castro, **Introdução à Geometria Projetiva e Cônicas Projetivas**, Notas de Aula.

Leandro Augusto Frata Fernandes, **Um Método Projetivo para Cálculo de Dimensões de Caixas em Tempo Real**, Dissertação

Teresa C. S. Azevedo¹, João Manuel R. S. Tavares e Mário A. P. Vaz, **Análise do Método de Calibração de Câmeras Proposto por Zhang**, Artigo

Eduardo Souza Mattos, **Geometria Projetiva e Visão Computacional**, Dissertação

Backes, André Ricardo; Sá Junior, Jarbas Joaci De Mesquita, **Introdução à Visão Computacional Usando MATLAB**

Hugo Mazon; Giancarlo Philippi Zacchi; Ricardo Martins, **Calibração de câmeras e fontes de erros para triangulação fotogramétrica**, Artigo

Jonas Gomes e Luiz Velho, **Fundamentos da Computação Gráfica**, IMPA, 2004