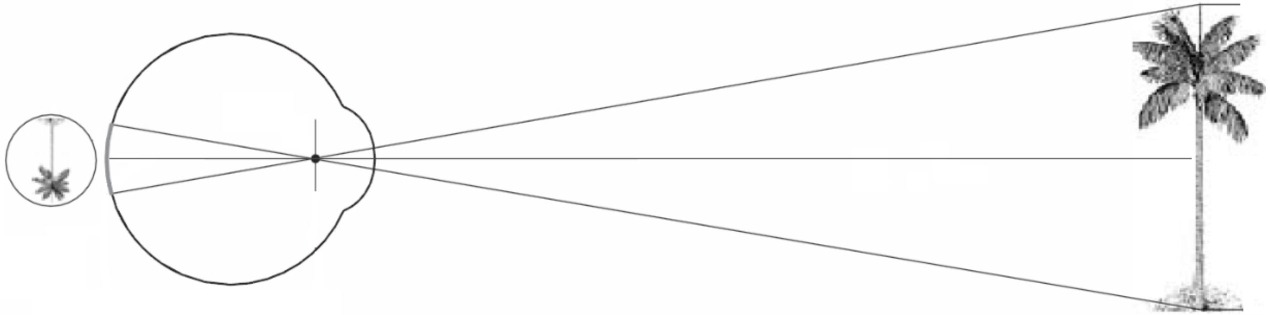
FUNDAMENTOS DE PROCESSAMENTO DE IMAGENS

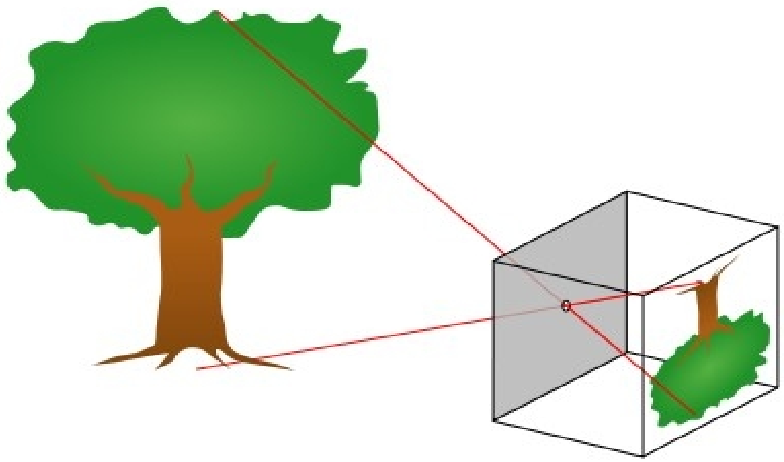
- Modelo de Câmera *Pinhole*
- A Geometria da Formação de Imagens

GEOMETRIA DE FORMAÇÃO DA IMAGEM

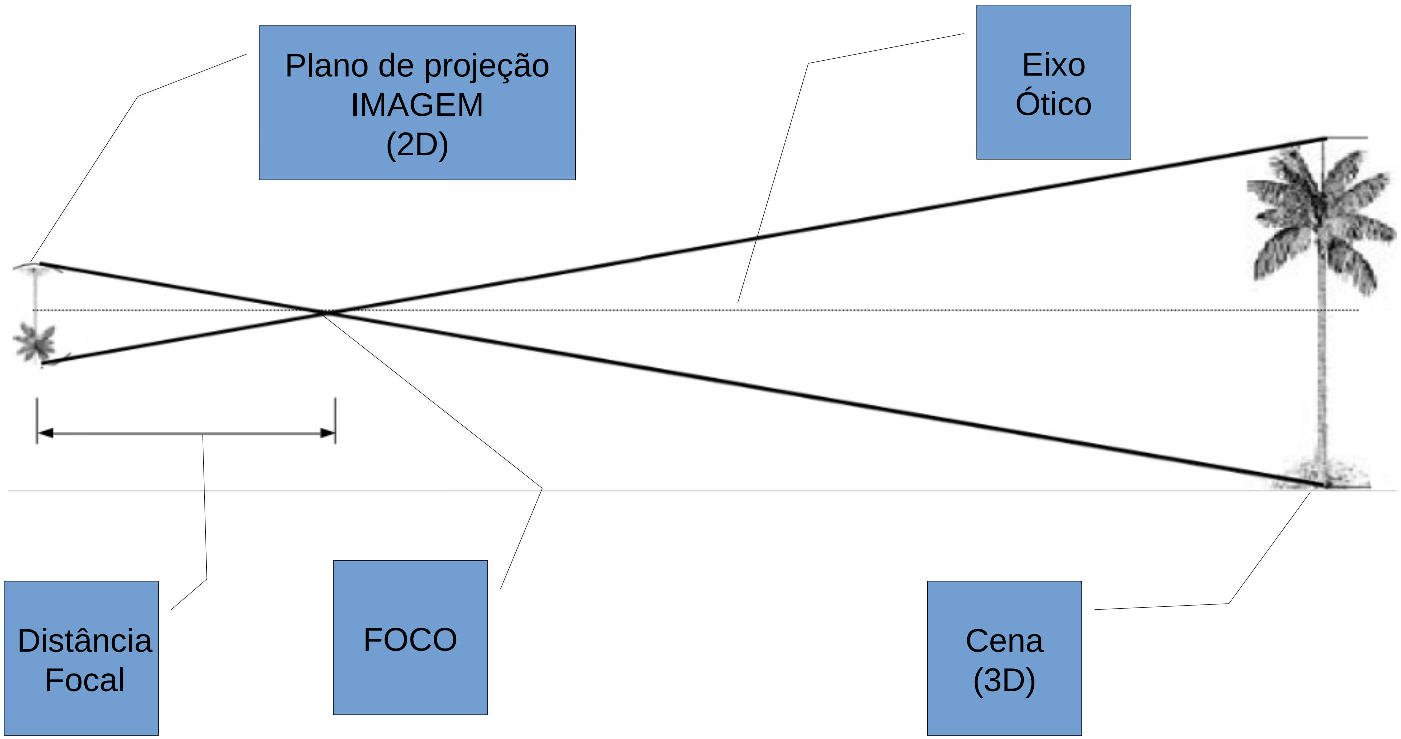
COM UM SISTEMA DE COORDENADAS NA CÂMERA

Formação de imagem na câmera de orifício (Pinhole)
Utilizaremos essa simplificação para deduzir um modelo matemático e projeção da cena sobre o sensor.

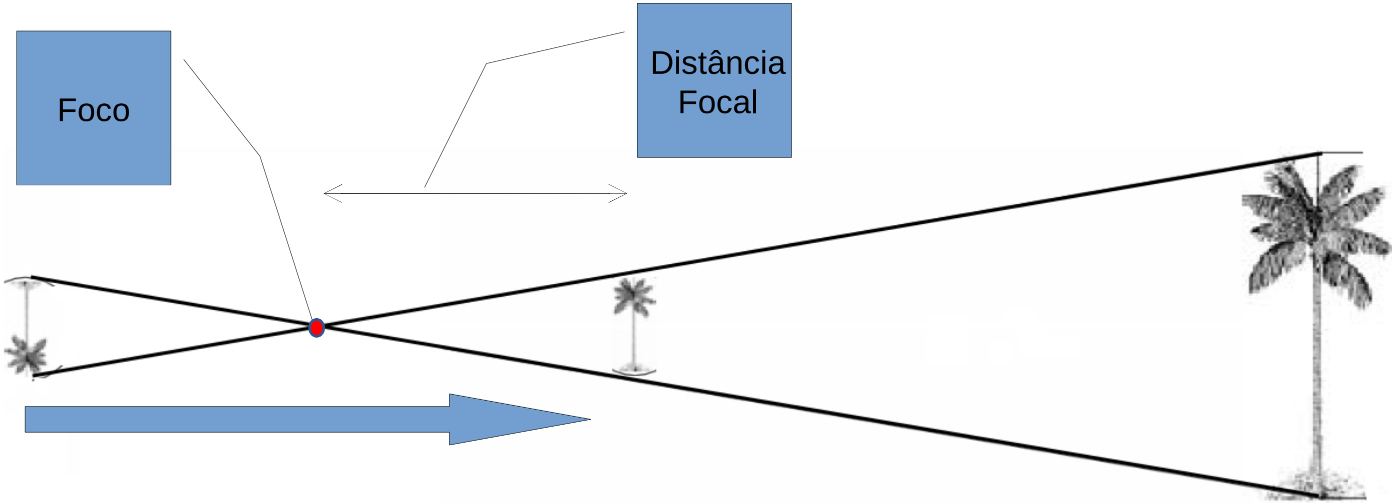




A geometria da formação de imagem na retina é similar à da câmera de orifício (Pinhole)

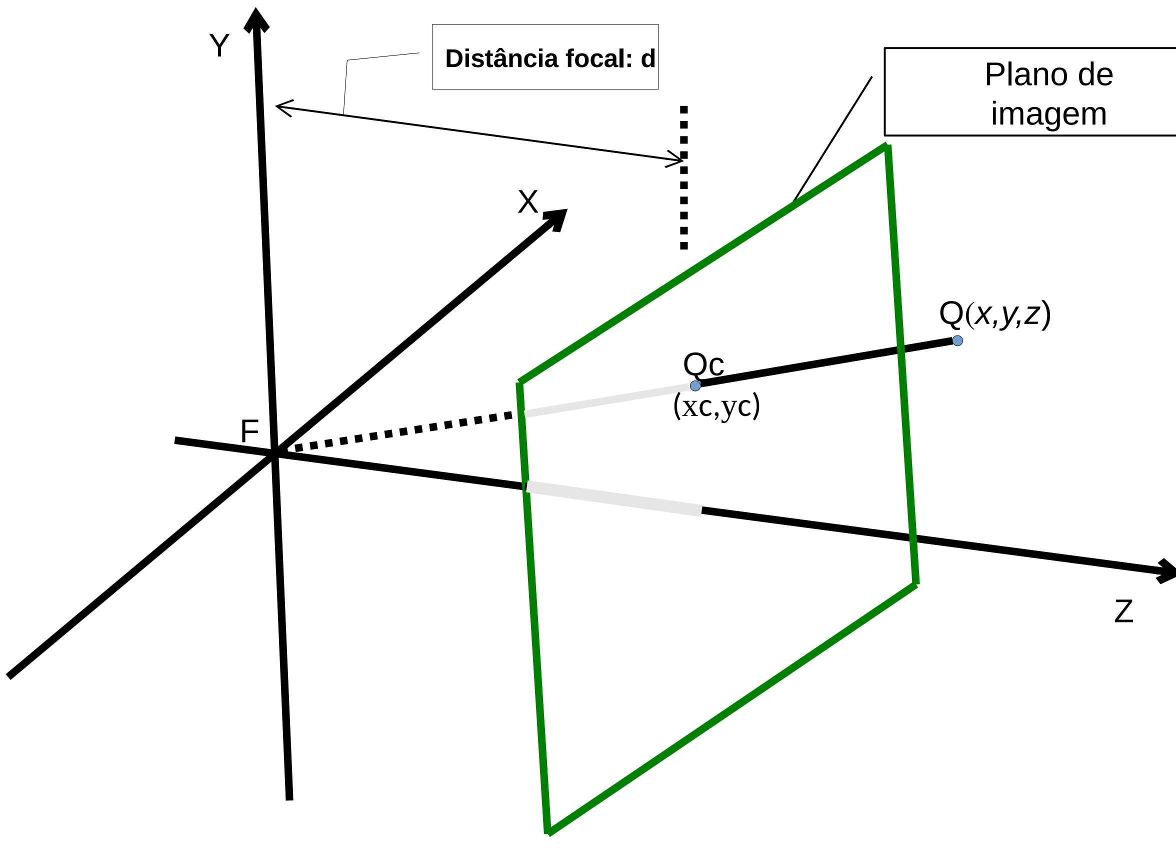


Formação de imagem na câmera Pinhole

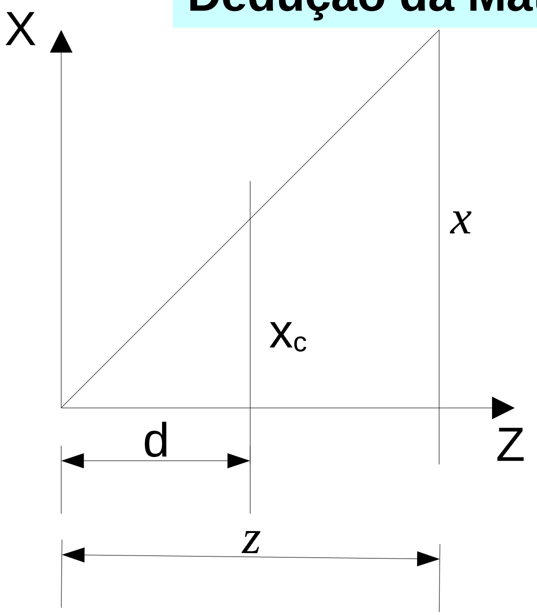


Por conveniência matemática, movemos o plano de imagem para a frente do foco.
Isso permite uma modelagem mais simples.

Projeção perspectiva: plano de projeção deslocado para a frente do centro de projeção sobre o eixo +Z

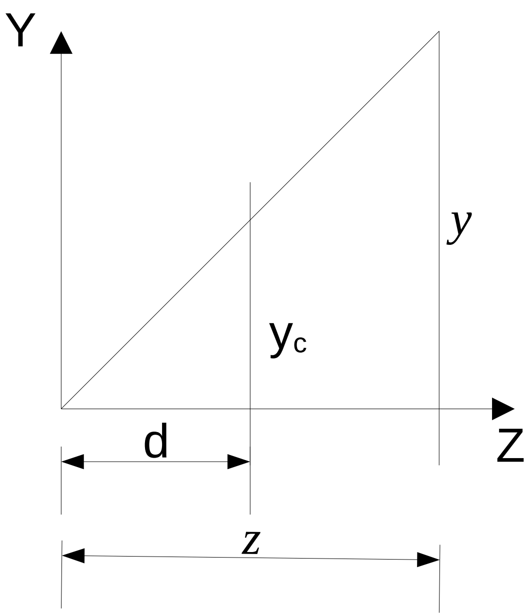


Dedução da Matriz de projeção perspectiva

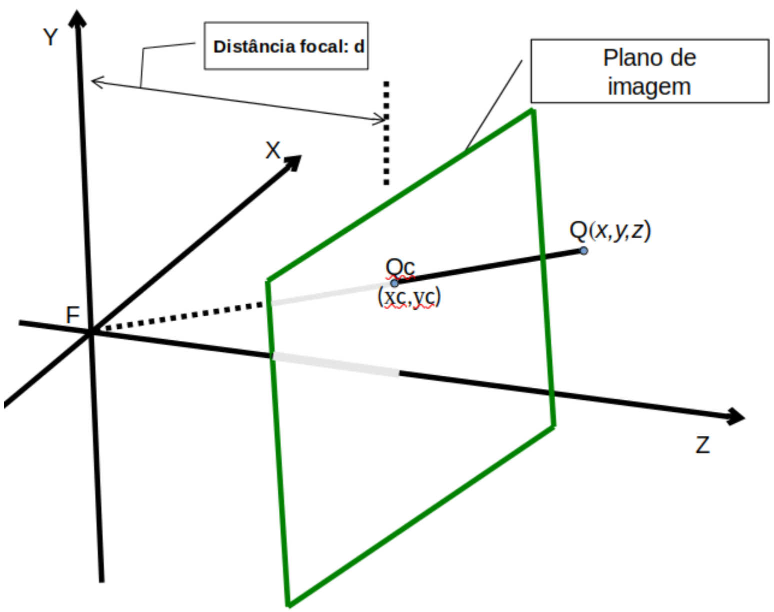


$$\frac{x_c}{d} = \frac{x}{z} \Rightarrow x_c = \frac{x}{z/d}$$

$$\frac{y_c}{d} = \frac{b}{c} \Rightarrow y_c = \frac{y}{z/d}$$



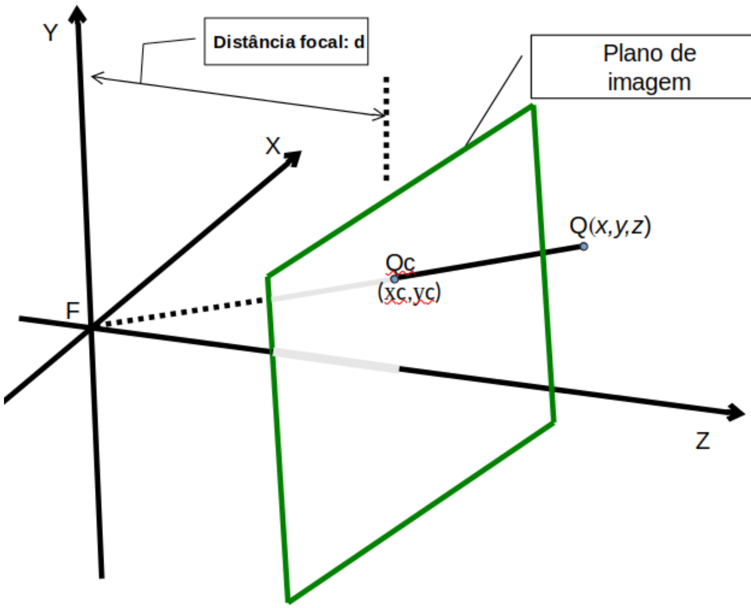
A) Matriz de projeção perspectiva considerando:
Plano de projeção normal ao eixo Z a uma distância *d* em relação a origem**.**



$$M_{per} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/d \end{bmatrix}$$

x_c e y_c:
são coordenadas da projeção de Q sobre a imagem.
estão em unidades métricas iguais às de Q (milímetros, por exemplo)

A) Matriz de projeção perspectiva considerando:
Plano de projeção normal ao eixo Z a uma distância *d* em relação a origem**.**



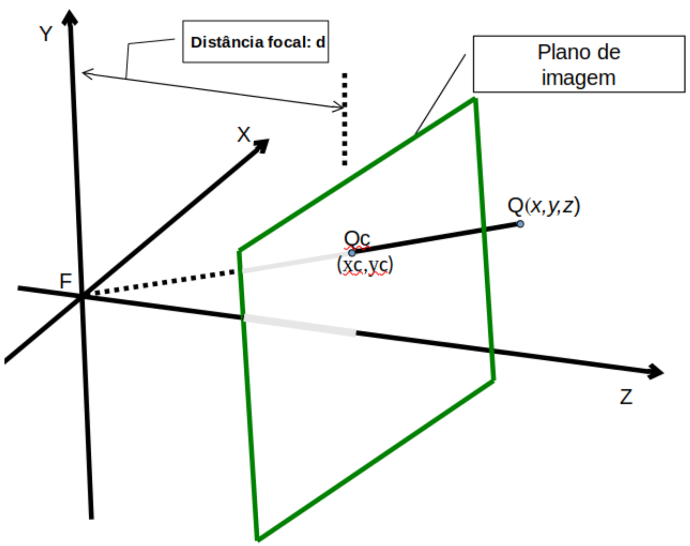
Q_h é o ponto no espaço homogêneo, com w=z/d

Q_c é o ponto no espaço cartesiano = Q_h/w

Q_p é o ponto em pixels

$$Q_h = M * Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/d \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x \\ y \\ z/d \end{bmatrix}$$

A) Exemplo



Q_h é o ponto no espaço homogêneo, com w=100

Q_c é o ponto no espaço cartesiano = Q_h/w

Q_p é o ponto em pixels

$$\text{d=5mm} \quad \text{Q=}\begin{bmatrix} 150,75 \\ 1500 \\ 500 \end{bmatrix} mm$$

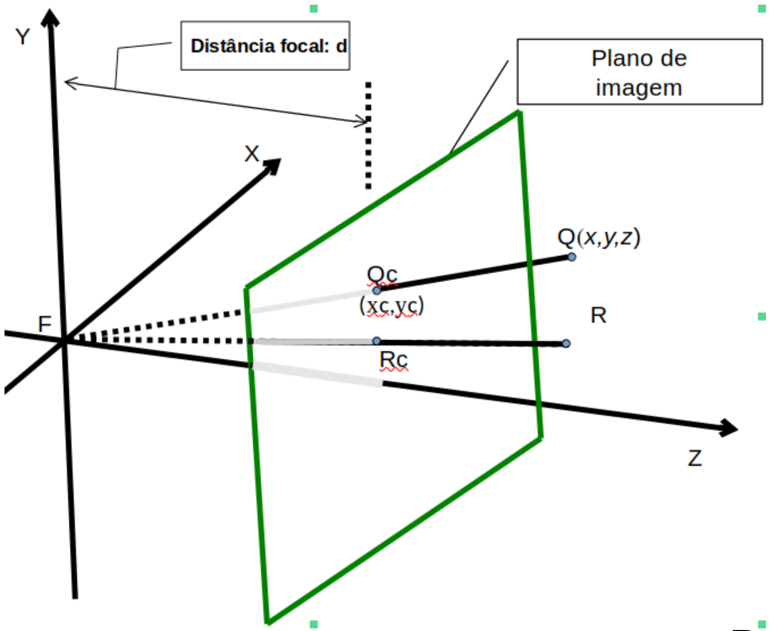
$$\mathrm{Q}_c = \begin{bmatrix} ? \\ ? \end{bmatrix} mm \;\; Q_p = \begin{bmatrix} ? \\ ? \end{bmatrix} \text{pixels}$$

$$\mathrm{Q}_h = M * Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/d \end{bmatrix} * \begin{bmatrix} 150,75 \\ 1500 \\ 500 \end{bmatrix} = \begin{bmatrix} 150,75 \\ 1500 \\ 100 \end{bmatrix} = \begin{bmatrix} x_h \\ y_h \\ w \end{bmatrix}$$

$$\mathrm{Q}_c = \begin{bmatrix} x_h/w \\ y_h/w \\ w/w \end{bmatrix} = \begin{bmatrix} 1,51 \\ 15 \\ 1 \end{bmatrix} mm$$

$$\mathrm{Q}_p = ???$$

A) Exemplo



$$Q=\begin{bmatrix}150,75\\1500\\500\end{bmatrix}mm \rightarrow \quad Q_c=\begin{bmatrix}X_c^Q=1,51\\Y_c^Q=15,0\end{bmatrix}mm$$

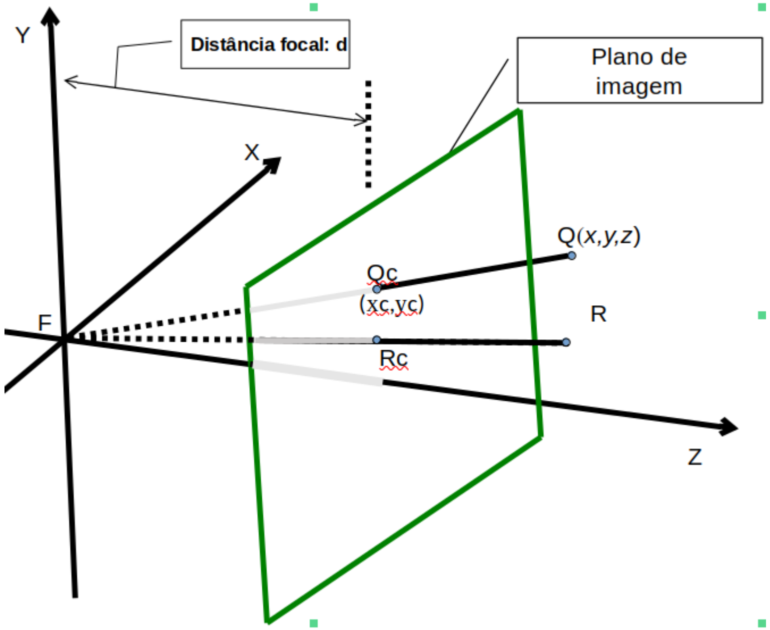
$$R=\begin{bmatrix}150,75\\1510\\500\end{bmatrix}mm \rightarrow \quad R_c=\begin{bmatrix}X_c^R=?\\Y_c^R=?\end{bmatrix}$$

$$R_h=M * R=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1/5\end{bmatrix} * \begin{bmatrix}150,75\\1510\\500\end{bmatrix}=\begin{bmatrix}150,75\\1510\\100\end{bmatrix}=\begin{bmatrix}X_h^R\\Y_h^R\\W\end{bmatrix}$$

$$X_h^R=150,75 \quad Y_h^R=1510 \quad W=100$$

$$R_c=\frac{R_h}{W}=\begin{bmatrix}X_h^R/W\\Y_h^R/W\\W\end{bmatrix}=\begin{bmatrix}1,51\\15,1\\1\end{bmatrix}\Rightarrow \quad X_c^R=1,51\,mm \quad Y_c^R=15,1\,mm$$

A) Exemplo



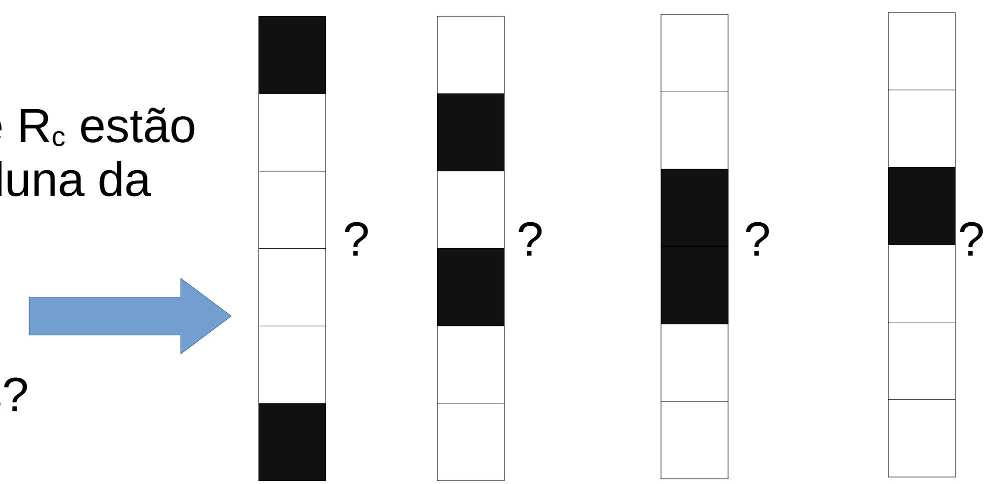
$$Q_c = \begin{bmatrix} X_c^Q \\ Y_c^Q \end{bmatrix} = \begin{bmatrix} 1{,}51\,mm \\ 15{,}0\,mm \end{bmatrix}$$

$$R_c = \begin{bmatrix} X_c^R \\ Y_c^R \end{bmatrix} = \begin{bmatrix} 1{,}51\,mm \\ 15{,}1\,mm \end{bmatrix}$$

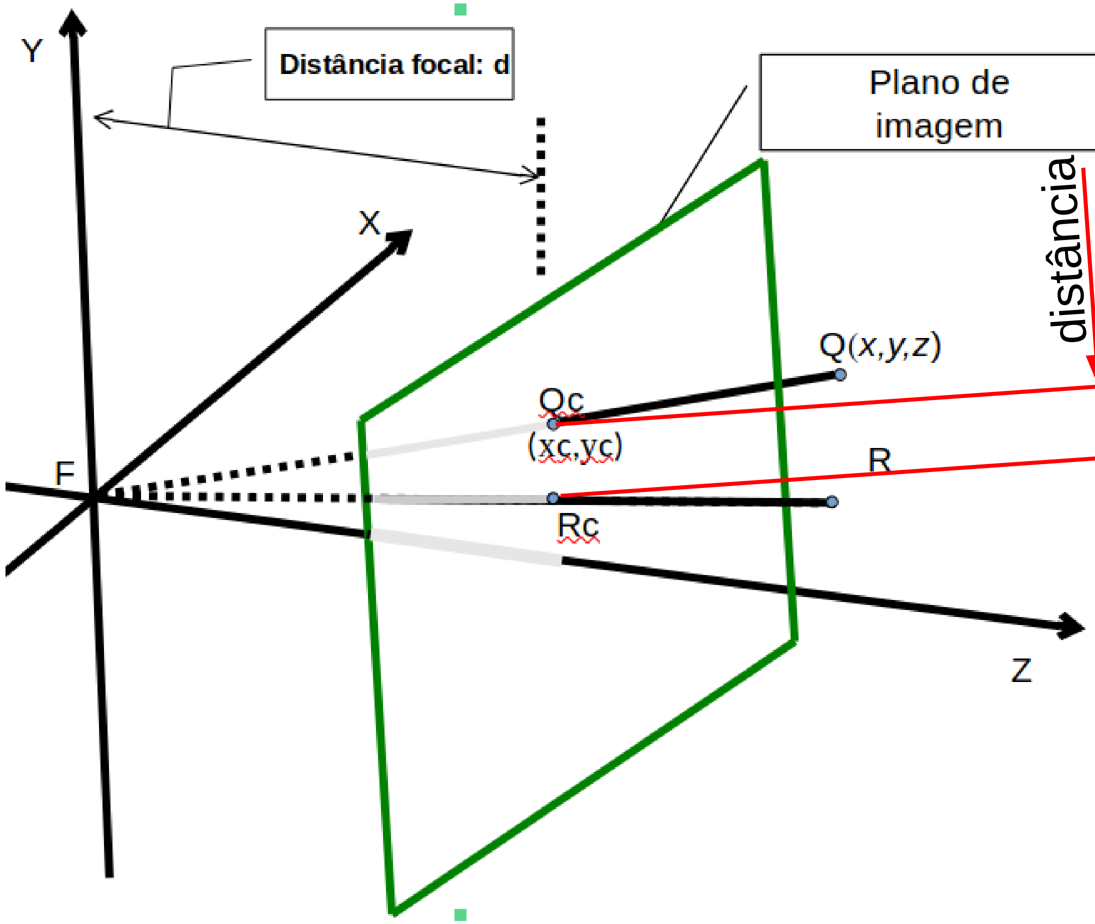
Se um pixel for um quadrado de 7,5 microns (0,0075 mm), será possível separar R de Q na imagem? A resposta depende de quantos pixels separam os pontos projetados.

No exemplo acima, Q_c e R_c estão alinhados na mesma coluna da Imagem.

Mas, qual seria o caso?
Há separação em pixels?



A) Exemplo



No exemplo, Q_c e R_c estão alinhados na mesma coluna da Imagem.

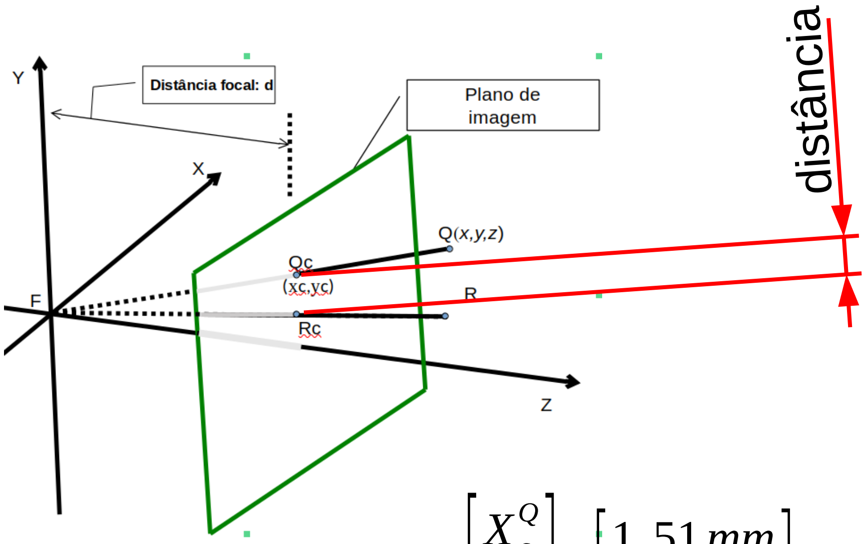
Como saber se Q_c e R_c estão separados?
Como calcular, em pixels, a separação ?

$$Q_c = \begin{bmatrix} X_c^Q \\ Y_c^Q \end{bmatrix} = \begin{bmatrix} 1{,}51\,mm \\ 15{,}0\,mm \end{bmatrix}$$

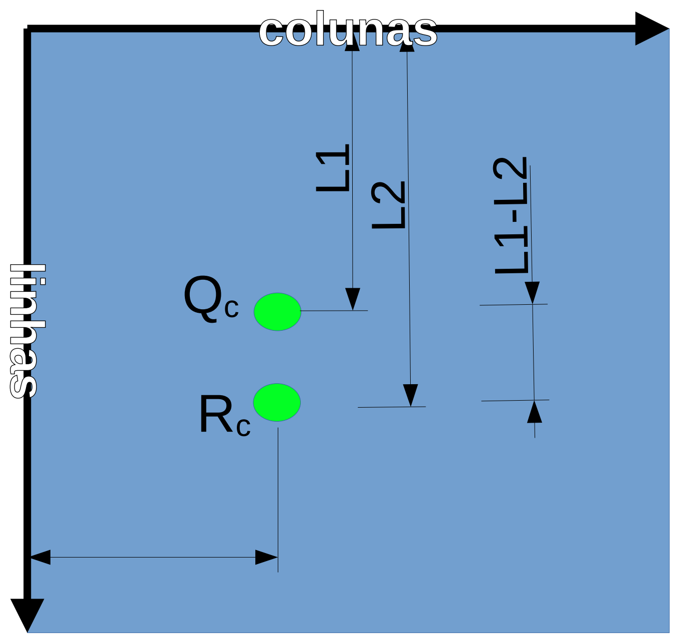
$$R_c = \begin{bmatrix} X_c^R \\ Y_c^R \end{bmatrix} = \begin{bmatrix} 1{,}51\,mm \\ 15{,}1\,mm \end{bmatrix}$$

Calculando a separação em pixels via Método Braçal

Calculando a separação em pixels via Método Braçal



A imagem é uma matriz de pixels referenciada no canto superior esquerdo



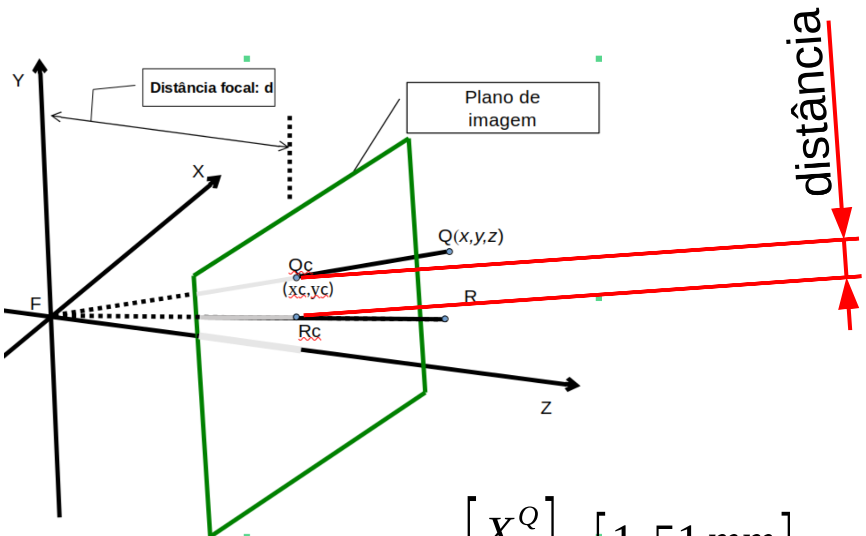
$$Q_c = \begin{bmatrix} X_c^Q \\ Y_c^Q \end{bmatrix} = \begin{bmatrix} 1{,}51\,mm \\ 15{,}0\,mm \end{bmatrix}$$

$$R_c = \begin{bmatrix} X_c^R \\ Y_c^R \end{bmatrix} = \begin{bmatrix} 1{,}51\,mm \\ 15{,}1\,mm \end{bmatrix}$$

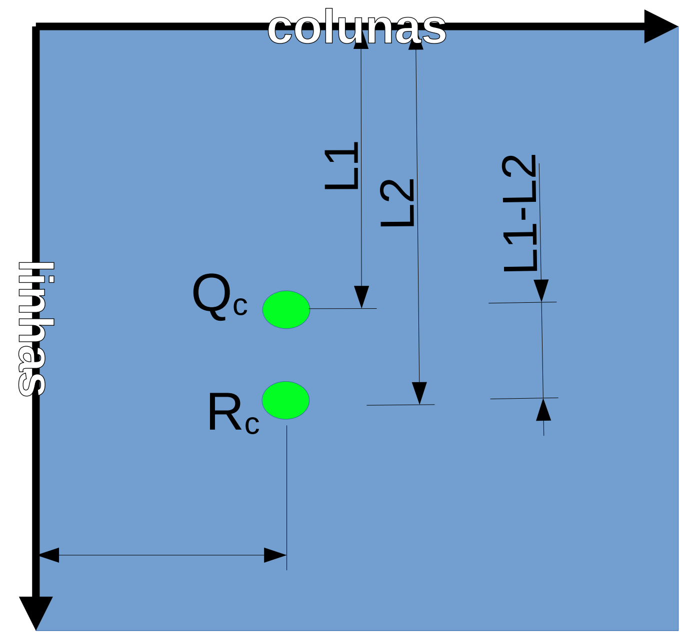
No caso, Q_c e R_c estão na mesma coluna, então a solução é calcular a distância L1-L2 em pixels e verificar se há separação.

Lembrando que a matriz M fornece a projeção em relação ao sistema XYZ na câmera, para determinar L1 e L2 é preciso fazer contas para conversão dos valores de unidades métricas (mm, microns...) para pixels… é preciso saber o tamanho do pixel!

Calculando a separação em pixels via Método Braçal



A imagem é uma matriz de pixels referenciada no canto superior esquerdo



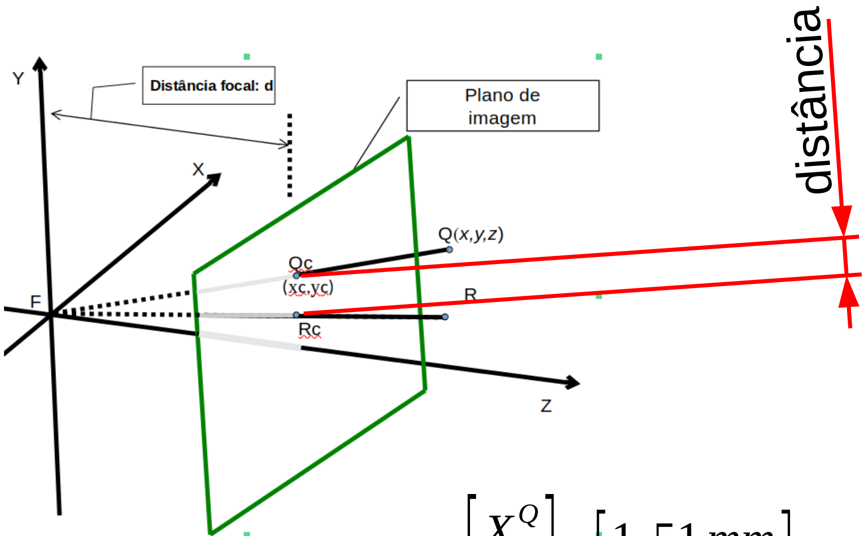
$$Q_c = \begin{bmatrix} X_c^Q \\ Y_c^Q \end{bmatrix} = \begin{bmatrix} 1{,}51\,mm \\ 15{,}0\,mm \end{bmatrix}$$

$$R_c = \begin{bmatrix} X_c^R \\ Y_c^R \end{bmatrix} = \begin{bmatrix} 1{,}51\,mm \\ 15{,}1\,mm \end{bmatrix}$$

Cálculo de L1 e L2 a partir das especificações do tamanho de pixel, por exemplo, para um pixel quadrado de lado = 0,0075 mm, teremos:

$$L_1 = \frac{15{,}0}{0{,}0075} = 2000 \quad L_2 = \frac{15{,}1}{0{,}0075} \approx 2013$$

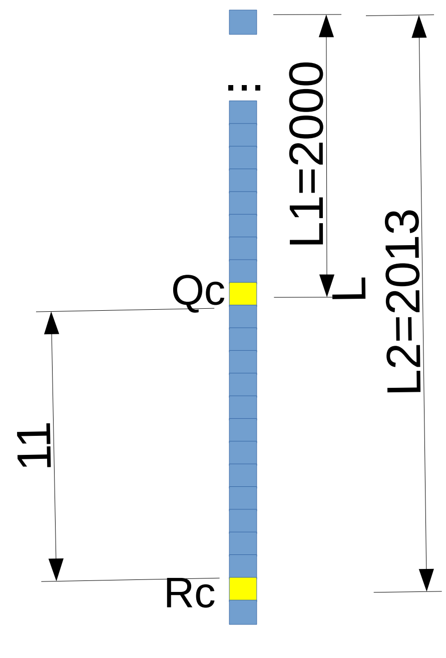
Calculando a separação em pixels via Método Braçal



$$Q_c = \begin{bmatrix} X_c^Q \\ Y_c^Q \end{bmatrix} = \begin{bmatrix} 1{,}51\,mm \\ 15{,}0\,mm \end{bmatrix}$$

$$R_c = \begin{bmatrix} X_c^R \\ Y_c^R \end{bmatrix} = \begin{bmatrix} 1{,}51\,mm \\ 15{,}1\,mm \end{bmatrix}$$

A imagem é uma matriz referenciada no canto superior esquerdo



L2-L1=13, portanto há 11 pixels de separação entre Q_c e R_c

$$L_1 = \frac{15{,}0}{0{,}0075} = 2000$$

$$L_2 = \frac{15{,}1}{0{,}0075} \approx 2013$$

Calculando a separação em pixels via Método Braçal

Exercício:

Suponha o caso abaixo para a mesma câmera discutida anteriormente, descreva a resolução de Q e R na imagem.
Ops pontos são separáveis? Qual é a distância em pixels entre eles?

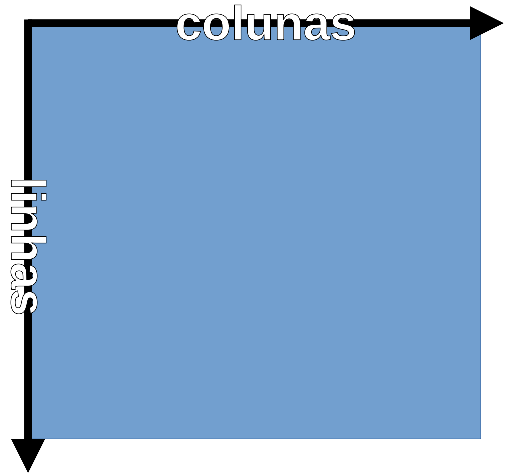
$$Q=\begin{bmatrix}150{,}75\\1500\\500\end{bmatrix}mm$$

$$R=\begin{bmatrix}180{,}75\\1510\\500\end{bmatrix}mm$$

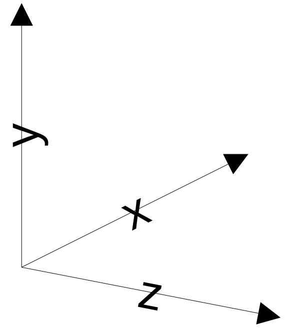
Calculando a separação em pixels via Método Eficiente – via implementação por produto matricial

Calculando a separação em pixels via Matriz de Transformação

A imagem é uma matriz de pixels referenciada no canto superior esquerdo



É preciso alinhar



Uma Modelagem mais Completa

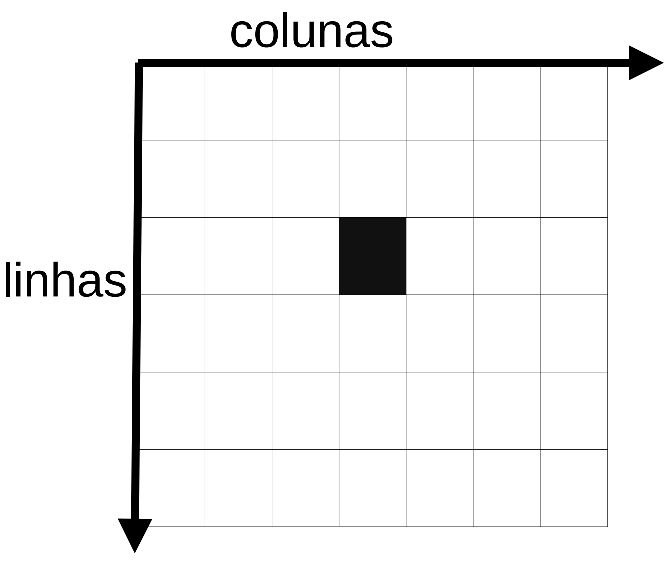
A modelagem apenas com a matriz M nos ajuda a entender o conceito de resolução de pontos projetados, porém, não fornece a melhor solução no que tange à obtenção das coordenadas em pixels de forma mais direta e de fácil implementação computacional.

Ao invés de fornecer o ponto projetado homogêneo em unidades métricas (mm, microns, etc) em relação ao sistema XYZ na câmera (o que não é intuitivamente associado a conceitos tais como o de resolução), seria muito mas eficiente se a matriz da transformação fornecesse o ponto projetado homogêneo, em pixels e respeitando a orientação do sistema matricial da imagem.

$$Q_h = M * Q = \begin{bmatrix} X_h \\ Y_h \\ W \end{bmatrix} pixels$$

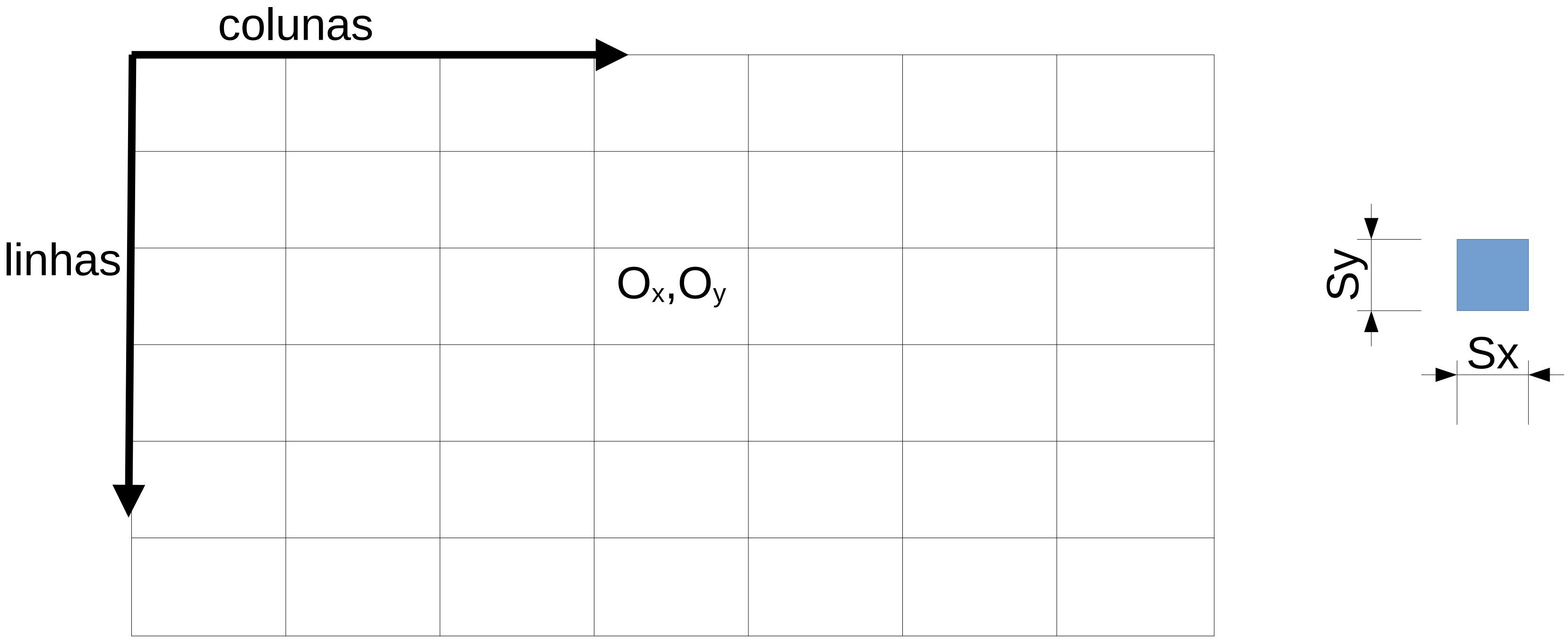
De onde facilmente se pode obter as coordenadas em linhas e colunas:

$$Q_p = \begin{bmatrix} X_h / W \\ Y_h / W \end{bmatrix} pixels = \begin{bmatrix} coluna \\ linha \end{bmatrix} pixels$$



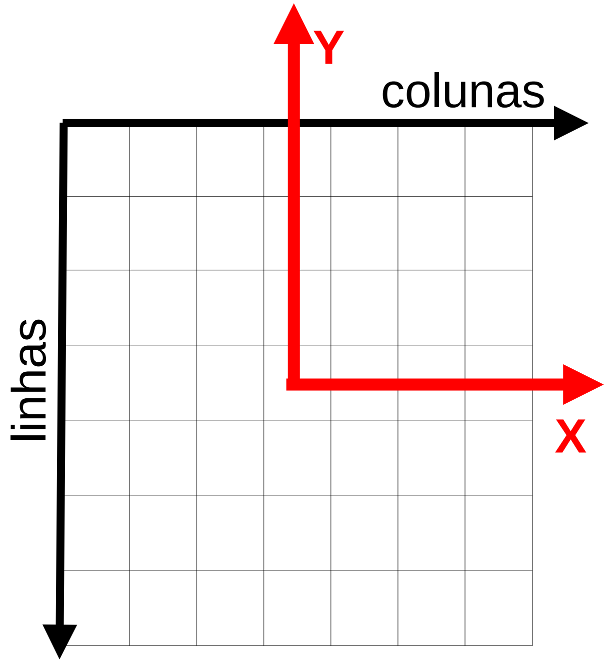
Uma Modelagem mais Completa

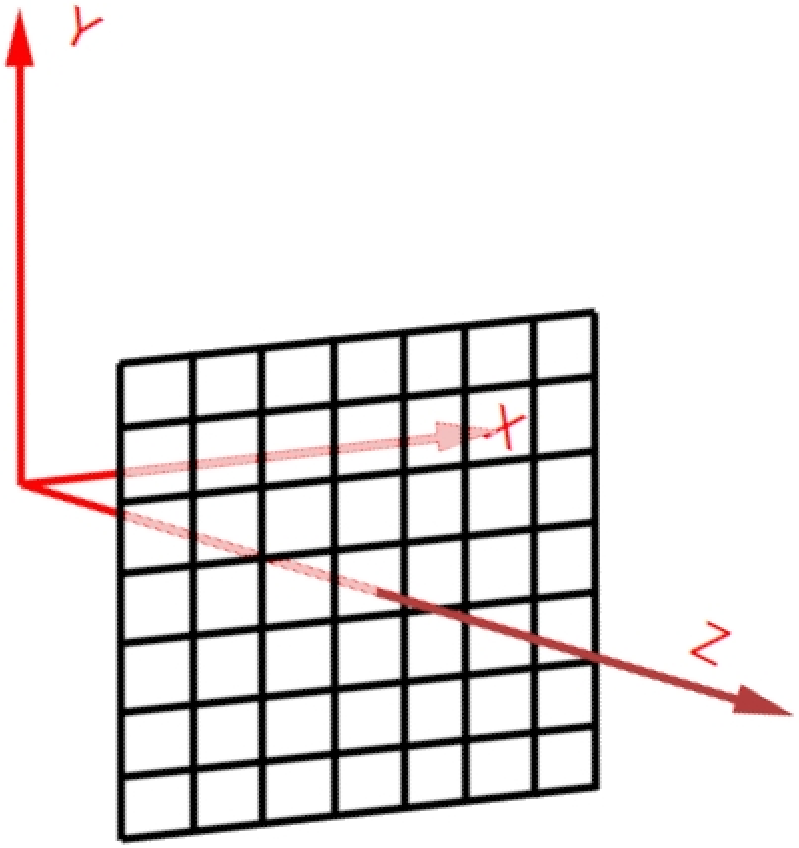
Essa nova matriz de transformação depende não apenas da distância focal da câmera, mas também das dimensões de pixel (s_x, s_y) e das coordenadas do centro da imagem (O_x,O_y) fornecidas em pixels, esses, juntamente com a distância focal são os parâmetros intrínsecos da câmera.



Uma Modelagem mais Completa

Para obter a transformação agora desejada é necessário aplicar transformações adicionais sobre a matriz M: um alinhamento do sistema X-Y com o sistema linhas-colunas e a utilização do tamanho efetivo do pixel.

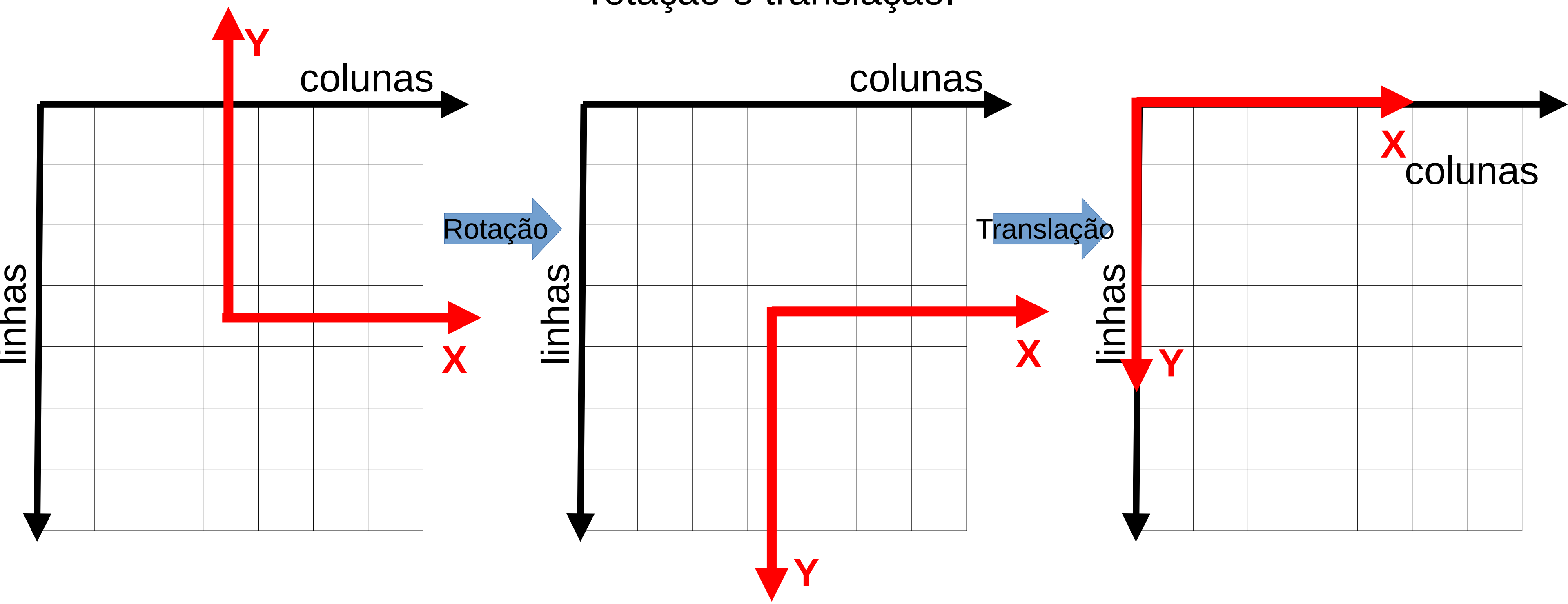




Uma Modelagem mais Completa

Para obter a transformação agora desejada, resumidamente, é necessário aplicar sobre a matriz M: um alinhamento do sistema X-Y com o sistema linhas-colunas e a utilização do tamanho efetivo do pixel.

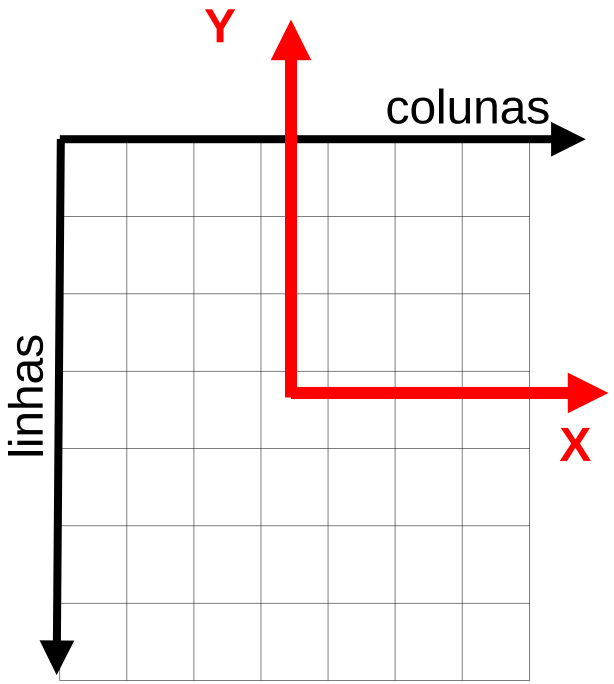
Para alinhar os dois sistemas são necessárias: rotação e translação.



Uma Modelagem mais Completa

A seguir temos o Passo-a-Passo para obter a nova matriz de projeção. Lembrando que originalmente temos:

$$Q_h = M * Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/d \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x \\ y \\ z/d \end{bmatrix} \quad \Rightarrow Q_c = \frac{Q_h}{z/d} = \begin{bmatrix} \frac{x}{z/d} \\ \frac{y}{z/d} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{d.x}{z} \\ \frac{d.y}{z} \\ 1 \end{bmatrix}$$

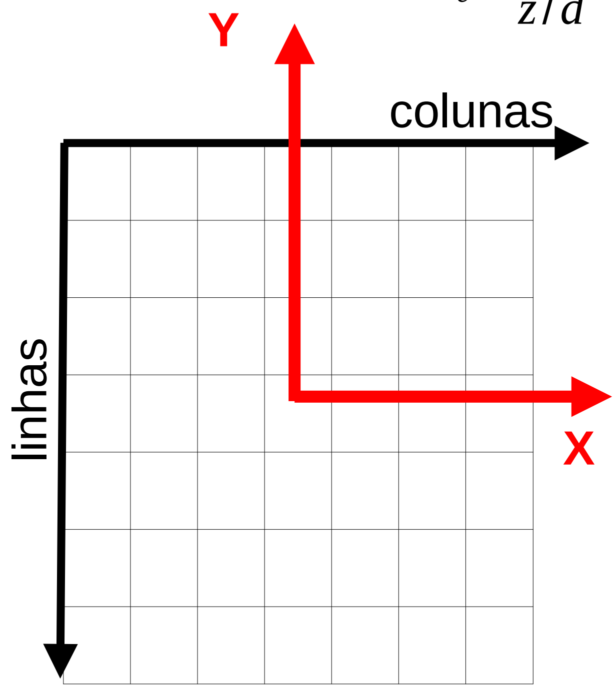


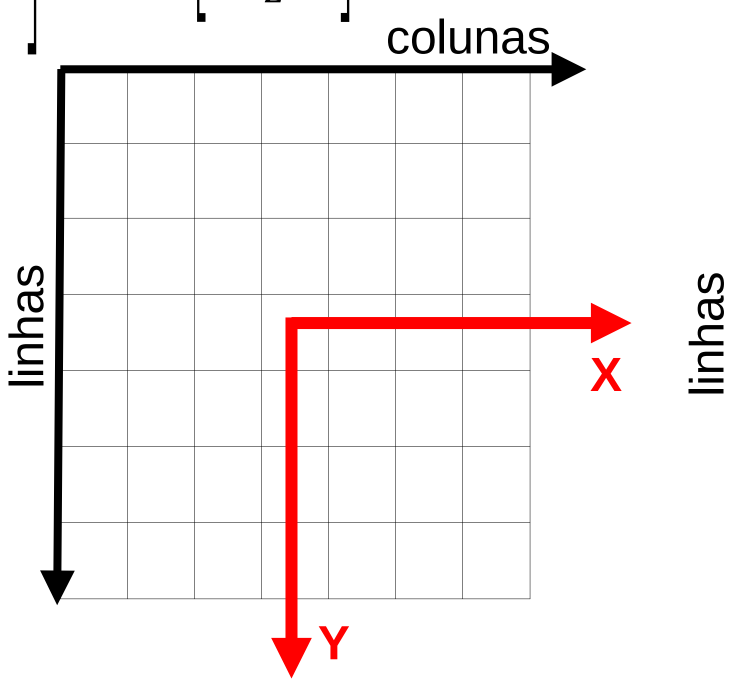
Uma Modelagem mais Completa

Rotação conveniente: alinha XY e linha-coluna

$$Q_h = Rot * M * Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/d & 0 \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ -y \\ z/d \end{bmatrix}$$

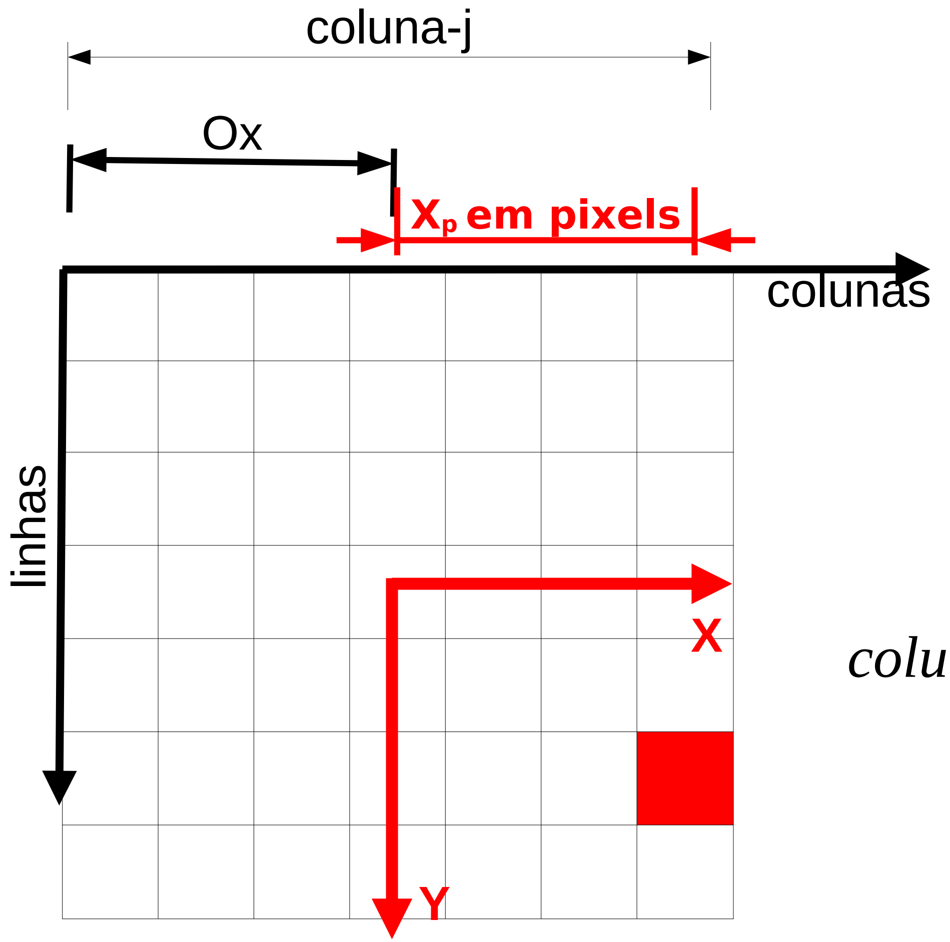
$$\Rightarrow Q_c = \frac{Q_h}{z/d} = \begin{bmatrix} \dfrac{d.x}{z} \\ \dfrac{-d.y}{z} \\ 1 \end{bmatrix} \equiv \begin{bmatrix} x_c \\ y_c \end{bmatrix} = \begin{bmatrix} \dfrac{d.x}{z} \\ \dfrac{-d.y}{z} \end{bmatrix}$$





Uma Modelagem mais Completa

$$Q_h = Rot * M * Q = \begin{bmatrix} x \\ -y \\ z/d \end{bmatrix} \quad \Rightarrow Q_c = \frac{Q_h}{z/d} = \begin{bmatrix} x_c \\ -y_c \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{d.x}{z} \\ \frac{-d.y}{z} \\ 1 \end{bmatrix}$$



Calculando as coordenadas da coluna em pixels :

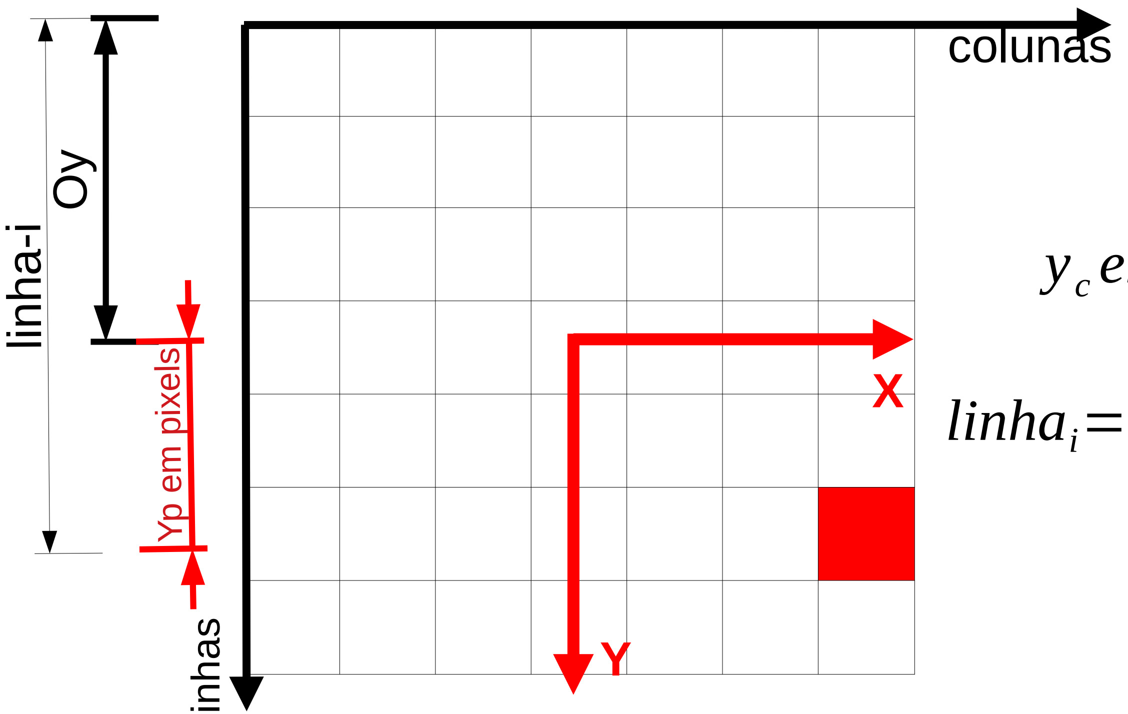
$$x_c \, em \, pixels = \frac{\left(\frac{d.x}{z}\right)}{s_x} = \frac{d.x}{z.s_x}$$

$$coluna_j = O_x + x_c \, em \, pixels = O_x + \frac{d.x}{z.s_x}$$

Uma Modelagem mais Completa

$$Q_h = Rot * M * Q = \begin{bmatrix} x \\ -y \\ z/d \end{bmatrix} \quad \Rightarrow Q_c = \frac{Q_h}{z/d} = \begin{bmatrix} x_c \\ -y_c \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{d.x}{z} \\ \frac{-d.y}{z} \\ 1 \end{bmatrix}$$

Calculando as coordenadas da linha em pixels :



$$y_c \, em \, pixels = \frac{\left(\frac{-d.y}{z}\right)}{s_y} = \frac{-d.y}{z.s_y}$$

$$linha_i = O_y + y_c \, em \, pixels = O_y + \left(\frac{-d.y}{z.s_y}\right)$$

Uma Modelagem mais Completa

$$coluna_j = O_x + x_p \, em\, pixels = O_x + \frac{d\,.\,x}{z\,.\,s_x}$$

$$linha_i = O_y + y_p \, em\, pixels = O_y + \left(\frac{-d\,.\,y}{z\,.\,s_y}\right)$$

O ideal é termos a formulação matricial

$$Q_h = Translaçao(O_x, O_y) * escala(s_x, s_y) * Rot * M * Q =$$

$$\begin{bmatrix} 1 & 0 & O_x \\ 0 & 1 & O_y \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1/s_x & 0 & 0 \\ 0 & 1/s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/d \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \frac{x}{s_x} + \frac{O_x\,.\,z}{d} \\ \frac{-y}{s_y} + \frac{O_y\,.\,z}{d} \\ z/d \end{bmatrix}$$

$$\Rightarrow Q_c = \frac{Q_h}{z/d} = \begin{bmatrix} \frac{d\,.\,x}{s_x\, z} + O_x \\ \frac{-d\,.\,y}{s_y\, z} + O_y \\ 1 \end{bmatrix} \equiv \begin{bmatrix} \frac{d\,.\,x}{s_x\, z} + O_x \\ \frac{-d\,.\,y}{s_y\, z} + O_y \end{bmatrix} = \begin{bmatrix} coluna_j \\ linha_i \end{bmatrix}$$

Uma Modelagem mais Completa

$$Q_h = Translaçao(O_x, O_y) * escala(s_x, s_y) * Rot * M * Q =$$

$$\begin{bmatrix} 1 & 0 & O_x \\ 0 & 1 & O_y \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1/s_x & 0 & 0 \\ 0 & 1/s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/d \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1/s_x & 0 & O_x/d \\ 0 & -1/s_y & O_y/d \\ 0 & 0 & 1/d \end{bmatrix}$$

O ideal é termos
a formulação matricial e
condensada em uma única
matriz: $M_{completa}$

Uma Modelagem mais Completa

$$coluna_j = O_x + x_c\, em\, pixels = O_x + \frac{d\,.\,x}{z\,.\,s_x} \qquad linha_i = O_y + y_c\, em\, pixels = O_y + \left(\frac{-d\,.\,y}{z\,.\,s_y}\right)$$

Aplicando a $M_{completa}$ sobre Q

$$Q_h = M_{completa} * Q$$

$$\Rightarrow Q_h = \begin{bmatrix} 1/s_x & 0 & O_x/d & 0 \\ 0 & -1/s_y & O_y/d & 0 \\ 0 & 0 & 1/d & 0 \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} \dfrac{x}{s_x} + \dfrac{O_x\,.\,z}{d} \\ \dfrac{-y}{s_y} + \dfrac{O_y\,.\,z}{d} \\ z/d \end{bmatrix}$$

Uma Modelagem mais Completa

$$coluna_j = O_x + x_c \, em\, pixels = O_x + \frac{d\,.\,x}{z\,.\,s_x} \qquad linha_i = O_y + y_c \, em\, pixels = O_y + \left(\frac{-d\,.\,y}{z\,.\,s_y}\right)$$

$$Q_h = M_{completa} * Q = \begin{bmatrix} \frac{x}{s_x} + \frac{O_x\,.\,z}{d} \\ \frac{-y}{s_y} + \frac{O_y\,.\,z}{d} \\ z/d \end{bmatrix}$$

$$\Rightarrow Q_p = \frac{Q_h}{z/d} = \begin{bmatrix} \frac{d\,.\,x}{s_x z} + O_x \\ \frac{-d\,.\,y}{s_y z} + O_y \\ 1 \end{bmatrix} \equiv \begin{bmatrix} \frac{d\,.\,x}{s_x z} + O_x \\ \frac{-d\,.\,y}{s_y z} + O_y \end{bmatrix} = \begin{bmatrix} coluna_j \\ linha_i \end{bmatrix}$$

Conferindo com os valores calculados em equações

Uma Modelagem mais Completa

$$Q_h = M_{completa} * Q$$

$$Q_h = \begin{bmatrix} 1/s_x & 0 & O_x/d \\ 0 & -1/s_y & O_y/d \\ 0 & 0 & 1/d \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1/0{,}0075 & 0 & 2048/5 \\ 0 & -1/0{,}0075 & 2048/5 \\ 0 & 0 & 1/5 \end{bmatrix} * \begin{bmatrix} 150{,}75 \\ 1500{,}0 \\ 500 \end{bmatrix} = \begin{bmatrix} 2{,}249\,x\,10^5 \\ 4{,}8\,x\,10^3 \\ 100 \end{bmatrix}$$

$$Q_c = Q_h / w \Rightarrow Q_c \approx \begin{bmatrix} 2249 \\ 48 \\ 1 \end{bmatrix}$$

$$R_h = M_{completa} * R$$

$$R_h = \begin{bmatrix} 1/s_x & 0 & O_x/d \\ 0 & -1/s_y & O_y/d \\ 0 & 0 & 1/d \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1/0{,}0075 & 0 & 2048/5 \\ 0 & -1/0{,}0075 & 2048/5 \\ 0 & 0 & 1/5 \end{bmatrix} * \begin{bmatrix} 150{,}75 \\ 1510{,}0 \\ 500 \end{bmatrix} = \begin{bmatrix} 2{,}249\,x\,10^5 \\ 3{,}466\,x\,10^3 \\ 100 \end{bmatrix}$$

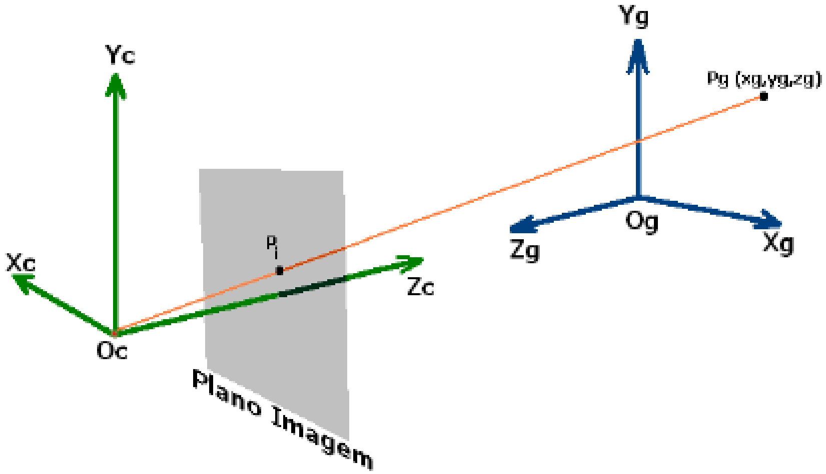
$$R_c = R_h / w \Rightarrow R_c \approx \begin{bmatrix} 2249 \\ 34 \\ 1 \end{bmatrix}$$

Supondo:
Ox=Oy=2048 pixels
sx=sy=0,0075 mm
D=5 mm

Independente de quaisquer das modelagens feitas anteriormente, o modelo representado na matriz M é restrito a um único sistemas de referência 3D na câmera. Na prática isso não é comum, ocorrendo um sistema na câmera e outro para os objetos da cena.

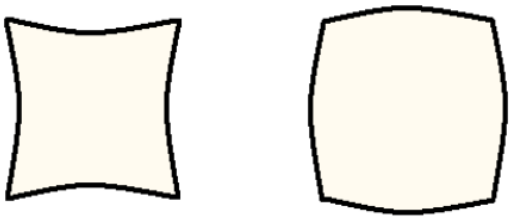
Nesse caso, além de levar em conta os parâmetros intrínsecos à câmera (tamanho de pixel, centro da câmera, distância focal) é necessário determinar a rotação e translação (parâmetros extrínsecos à câmera) para alinhamento dos dois sistemas 3D.

M’= (translação(Ox,Oy)*escala(sx,sy)* rot) * (Rotação*Translação)



E quanto a distorções por defeitos na lente?

Se forem significativas, podem ser tratadas por técnicas de correção



Pincushioning (almofada de alfinete)

Barrel (Barrilete)

Segue...