

# Trabalho de Implementação 1 — Visão Computacional

Artur Waquil Campana (287677) e Nicolás Casagrande Duranti (287679)

Implementamos o nosso trabalho em Python, usando as bibliotecas Numpy e OpenCV. A seleção de pontos de calibração foi feita manualmente, escolhendo locais estratégicos do campo — mais detalhes são dados em cada questão. Implementamos o método *Direct Linear Transformation* (DLT) em suas versões 2D e 3D. Para isso, consideramos que campo de futebol do estádio Maracanã tem dimensões de acordo com a padronização da CBF: 105m de comprimento e 68m de largura.

## Questão 1

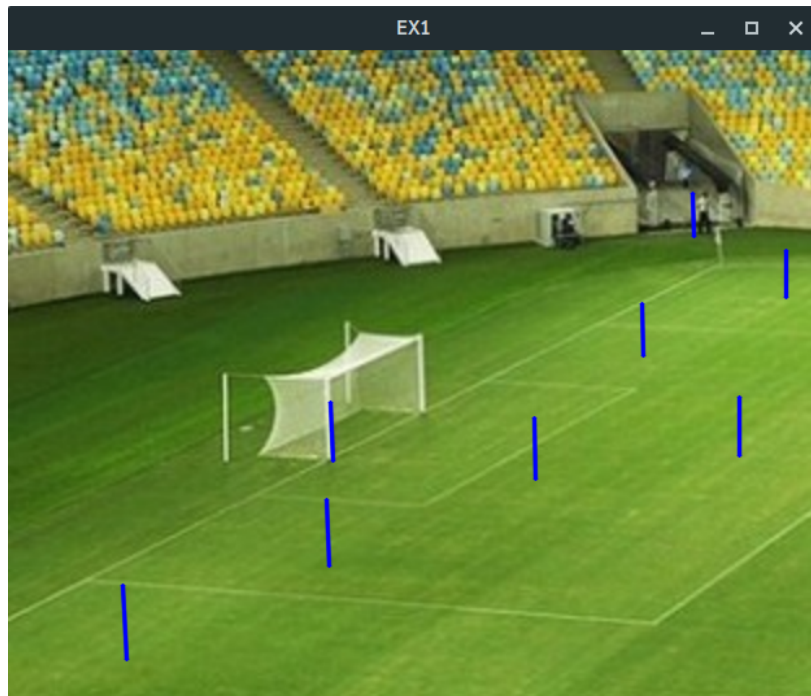
Para a questão 1, selecionamos os pontos ilustrados na Figura 1. As coordenadas foram obtidas manualmente. Aqui usamos a versão 3D da DLT, já que havia pontos fora do plano do campo (pontos C e D, na goleira).



Figura 1: pontos de calibração para a questão 1.

A partir da posição do cursor na tela ( $\text{cursor}_x$ ,  $\text{cursor}_y$ ) quando o mouse é clicado, encontramos a coordenada de mundo ( $\text{pés}_x$ ,  $\text{pés}_y$ ) correspondente aos pés do jogador. Para isso, encontramos a inversa da matriz  $P$  de câmera (a qual é possível obter ignorando a coluna respectiva ao eixo  $Z$ , já que os pés estão no plano do campo) e a multiplicamos pela coordenada de tela do cursor.

Conhecendo a coordenada de mundo dos pés, encontramos a coordenada de tela correspondente à cabeça do jogador multiplicando  $P$  pelo vetor  $(\text{pés}_x, \text{pés}_y, 1.80, 1)^T$ . Para desenhar o jogador, é impresso um segmento de reta entre os pontos  $(\text{pés}_x, \text{pés}_y)$  e  $(\text{cabeça}_x, \text{cabeça}_y)$  na tela. Na Figura 2, podem ser vistos alguns “jogadores” desenhados em diferentes locais no campo.



*Figura 2: alguns exemplos de jogador desenhados no campo.*

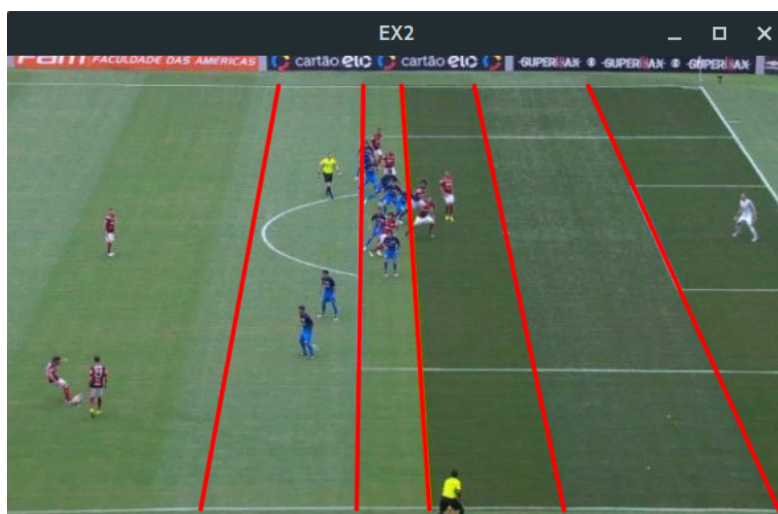
## Questão 2

Nesta questão, selecionamos os pontos ilustrados na Figura 3. Todos esses pontos de calibração se situam no plano do campo, por isso utilizamos a DLT 2D. A matriz  $P$  retornada pelo método bidimensional é inversível.



*Figura 3: pontos de calibração para a questão 2.*

Multiplicamos  $P^{-1}$  pela coordenada de tela do ponto clicado para obter a coordenada de mundo correspondente. Usando o valor  $y$  da coordenada de mundo e valores de  $x$  pertencentes às laterais do campo, encontramos as coordenadas de tela que definem as extremidades da linha a ser desenhada. Na Figura 4, podem ser observadas algumas linhas desenhadas; a linha central corresponde à linha de impedimento no lance.



*Figura 4: alguns exemplos de linhas de impedimento (a central é a linha de impedimento).*

## **Conclusão**

De maneira geral, nossos resultados correspondem ao que esperávamos. Na questão 1, a grande quantidade de pontos permitiu uma calibração pouco ruidosa, com os “jogadores” sendo projetados sem grandes distorções. Na questão 2, mesmo usando apenas os pontos no plano do campo, conseguimos uma boa calibração, com um ruído muito pequeno (que pode ser observado nas laterais do campo, na Figura 4).