

# Visualização científica de dados com OpenGL

**Pedro Henrique de Almeida Alonso**

Instituto Federal– Instituto Federal São Paulo (IFSP)

Presidente Epitácio – SP – Brazil

## 1. Introdução

OpenGL é uma API (Application Programming Interface) livre altamente utilizada para a computação gráfica, devido ao fornecimento de uma grande diversidade de recursos para a criação e manipulação de interfaces.

Este trabalho visa utilizar das funcionalidades disponíveis pelo OpenGL em conjunto com a linguagem de programação C++ para o desenvolvimento de uma ferramenta capaz de realizar a visualização científica de dados.

Originalmente a base de dados, disponibilizada em um arquivo CSV (Comma-separated values), a ser utilizada consistia de dados de 1034 jogadores de baseball da MLB, embora a base de dados não possui nenhuma data que informe o ano, uma pesquisa comparando a idade contida nesta, referente a alguns jogadores, com sua idade atual permitiu determinar que ela foi formada por volta de 2007.

A base contém dados diversos como nome do jogador, sigla do time o qual ele pertence, sua posição no campo, altura, peso e idade. Observando que dados referentes a altura e peso estavam em polegadas e libras, foi-se utilizado um programa em python3 o qual converteu os dados de libras para quilogramas e polegadas para centímetros. Assim como filtrou os jogadores baseado em times, para que apenas membros dos times BAL (Baltimore Orioles), CWS (College World Series), ANA (Santa Ana Baseball), BOS (Boston Red Sox) e CLE (Cleveland Indians) ficassem presentes no arquivo final.

Além do filtro em relações aos times, também foi-se aplicado um outro filtro para generalizar as posições dos jogadores no time, por exemplos jogadores que possuem dados como “Starting Pitcher” e “Relief Pitcher” passaram a ser apenas “Pitcher”. Através dessa posição generalizada o programa de visualização determinará o formato do ponto a ser utilizado.

## 2. Desenvolvimento

### 2.1 Processamento de dados

O projeto foi iniciado com a padronização e filtro dos dados presente no arquivo CSV referente aos jogadores do MLB de 2007, para isso foi desenvolvido um programa em python3 o qual utiliza da biblioteca CSV, para realizar a leitura, processamento e escrita do novo arquivo processado.

### 2.2 Armazenamento de dados

Utilizando deste arquivo processado, o programa irá ler linha por linha o arquivo identificando os dados de forma que utilizará a altura no eixo X, o peso no eixo Y e a idade do jogador no eixo Z. No entanto, este conjunto de dados não está padronizado de maneira que preencha corretamente o gráfico, para corrigir isto foi-se utilizado uma adaptação da função map presente no sistema de dispositivos embarcados como Arduino para mapear um conjunto de dados em determinado intervalo para outro. Ou seja, ela converterá os dados do intervalo referente ao menor e maior peso/altura para o intervalo referente aos pontos cartesianos do sistema (-2.5, 2.5).

### 2.3 Eixos

Em seguida procedemos com a criação de uma nova tela através do glfw3 e a criação dos eixos pertencentes ao sistema tridimensional de coordenadas, para isso foi determinado um ponto em uma localização que centralize o gráfico na janela.

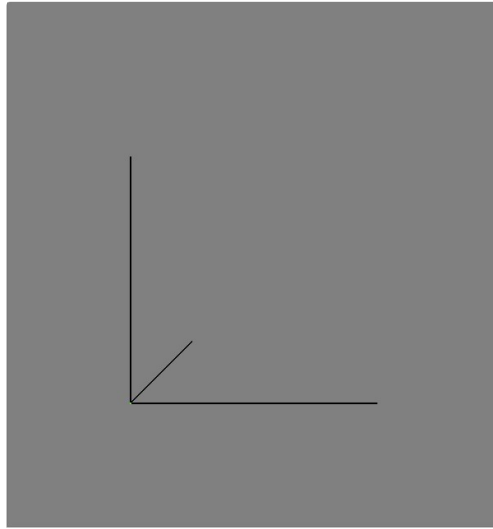


Figura 1: Desenho dos eixos

### 2.4 Grid de orientação

Após o desenho dos eixos que serão utilizados para a orientação da visualização do gráfico, desenvolvemos um conjunto de cubos para facilitar a identificação das posições dos pontos. Para isso o sistema foi subdividido em 10 repartições, as quais levam em consideração todos os eixos.

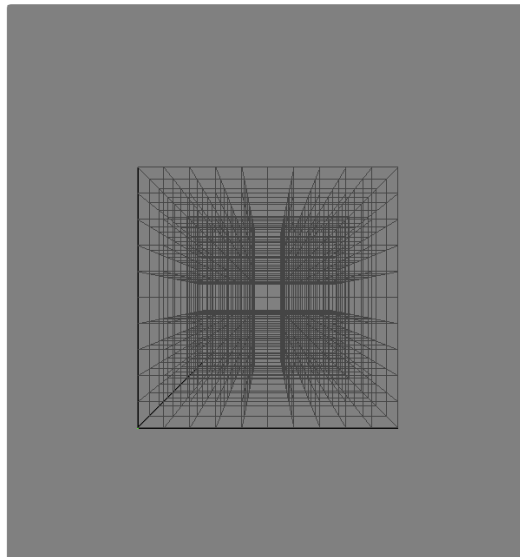


Figura 2: Desenho de cubos para facilitar a identificação das coordenadas dos pontos

## 2.5 Pontos

Utilizando-se das informações referentes a posição do jogador em campo, é determinado o tipo do ponto que será utilizado para que este seja representado no sistema. Para isso será utilizado diversas formas geométricas como pontos, círculos, esferas, cubos, pirâmides e triângulos. A qual possuirá a posição do seu ponto central determinado pela conversão de dados realizados pela função map. Embora isto cause problemas, com algumas formas geométricas serem desenhadas além das bordas do sistema, uma alteração para corrigir esse problema causaria um possível alteração nos dados ou prejudicaria a interpretação do gráfico, devido ao recorte de parte da forma.

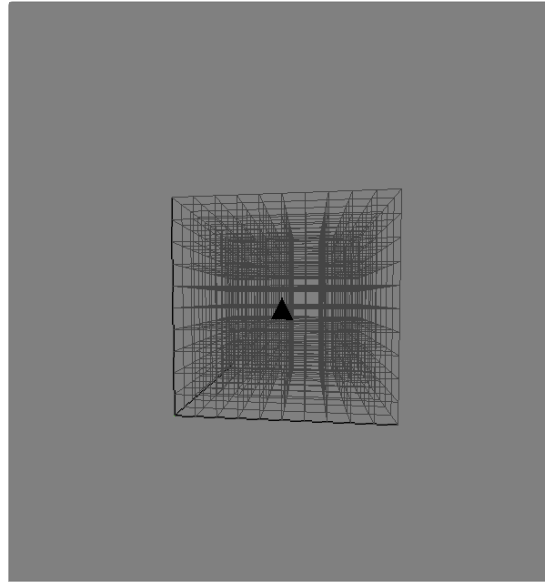


Figura 3: Desenho dos pontos utilizando símbolos geométricos

## 2.6 Interação e resultados

Com todas as etapas anteriores concluídas, ao adicionarmos uma função responsável por percorrer a lista que contem os dados dos jogadores e inseri-las no sistema, já é obter ter uma visualização científica dos dados.

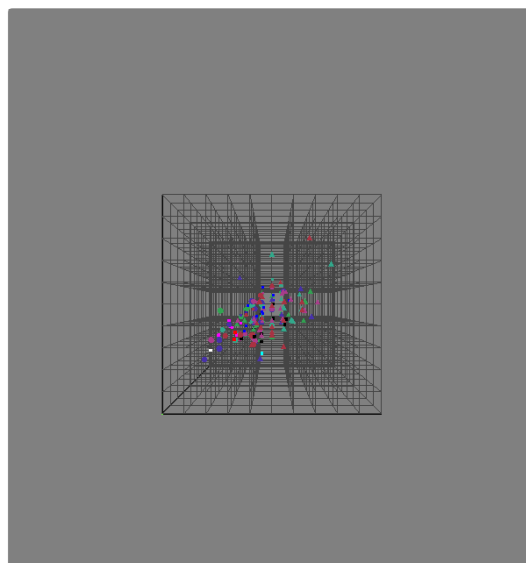


Figura 4: Visualização de dados

No entanto, nota-se dificuldade para manipular a visualização. Portanto foi implementado um sistema de controle da visão utilizando o click do mouse, o qual permite o usuário movimentar a câmera enquanto ele pressionar o botão esquerdo. Também há a possibilidade de manipular o zoom da câmera através do scroll do mouse, e aumentar a escala dos pontos ao utilizar os símbolos de adição (+) e subtração (-) presentes no teclado numérico.

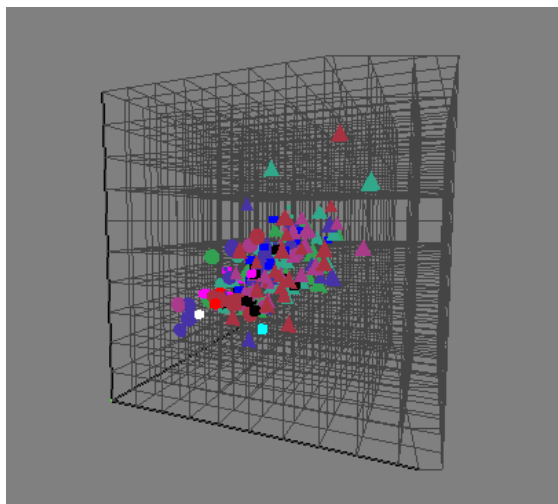


Figura 5: Manipulação da câmera, zoom da imagem e escala dos pontos

### 3. Conclusão

Com o desenvolvimento deste trabalho podemos concluir que o OpenGL é uma ferramenta que disponibiliza um grande nível de liberdade para que o usuário possa realizar o desenvolvimento de uma solução para o seu problema. Em conjunto com a tecnologia, existem diversas ferramentas, mecanismos e comportamentos que auxiliaram no desenvolvimento da ferramenta de visualização. Dentre elas se destacaram a capacidade de interagir, manipular e inserir novos objetos/pontos no ambiente gráfico, assim como não precisar se preocupar com a utilização de mais de duas dimensões, pois através da visão ortogonal o OpenGL utiliza da posição do objeto para manipular o tamanho que este será desenhado, dando assim uma sensação de profundidade.

Em relação a ferramenta desenvolvida, há a necessidade de adequar a manipulação da câmera através do arrastar do mouse, enquanto o botão esquerdo está pressionado, pois este movimento não demonstra uma sensação de “liso” e apresenta travamentos constantes ou imprecisão.

Outro problema identificado, o qual provavelmente está sendo causado mais por aspectos da máquina utilizada para o desenvolvimento deste trabalho se trata de congelamentos ocasionais quando se utiliza as teclas do teclado mecânico, funcionalidade a qual não demonstrou problemas em outros dispositivos.

Em questões dos dados utilizados, foi possível observar que estes se aglomeram no centro do gráfico, principalmente por representar os atributos considerados presentes em grande parte dos jogadores de baseball, no entanto esta decisão foi tomada sobre um gráfico gerado com times filtrados. Para que esta análise seja expandida dos 174 jogadores para todos os 1033, visa a implementação de uma tabela hash a qual armazenará a sigla do time e um código r, g e b o qual representa a cor que este time possuirá no gráfico. Para que dessa forma possamos continuar expandido a quantidade de jogadores representados e seus mais diversos times.