

PROJETO INTERATIVO III ANGRY ROBOTS

Carolina Bomfim, Mahaira Soares de Souza,
Rafael da Silva Santos e Thiago Messias

caroline.bomfim@hotmail.com.br, mahaira_souza@hotmail.com,
rafa_silva.santos@hotmail.com, messiassthi@gmail.com

"Angry Robots, é um jogo baseado em visão computacional, desenvolvido em linguagem C, versão 99, utilizando a biblioteca allegro 5, que possibilitou criar a interface gráfica do jogo. A principal interação do mesmo é através de uma câmera, utilizando a biblioteca OpenCV para desenvolver estes aspectos. Apresentado para a conclusão da disciplina, Projeto Interativo III, do bacharelado em Ciência da Computação, Centro Universitário Senac."

Sob orientação do Prof.^o: Marcelo Hashimoto.

RESUMO

Angry Robots, jogo originalmente criado por jovens do 3º semestre de Ciência da Computação, desenvolvido no decorrer do semestre proposto na disciplina de Projeto Interativo, com bibliotecas como OpenCV - "biblioteca totalmente multiplataforma, livre para o âmbito acadêmico e comercial também, utilizado para desenvolver aplicações em visão computacional", allegro - "biblioteca gráfica de código aberto com principal objetivo de games 2D". Atualmente existem muitos jogos no mercado cuja a interação é através de câmeras, desde jogos de dança, até jogos de aventura, neste programa em específico os algoritmos essenciais foram calculo de mediana, conversão para escala de cinza, segmentação e corte e centroide.

Palavras-chave: algoritmo, OpenCV, allegro, biblioteca gráfica.

ABSTRACT

Angry Robots, game originally created by young people from the 3rd semester of Computer Science, developed during the semester proposed in discipline Interactive Project with libraries like OpenCV - "totally multiplatform library, free for academic and business scope also used to develop applications in computer vision ", allegro - "open source graphics library with main objective of 2D games. Nowadays there are a lot of games that the main interaction is through of cameras, there are games about dance, adventure and so on. In this program the main algorithms was calculate median, converting to garyscale, cutting and threading, and many others."

Keywords: *algorithm, OpenCV, allegro, graphics library.*

INTRODUÇÃO

Segundo pesquisas[4], jogos que necessitam de movimentos rápidos e de raciocínio lógico por meio da interatividade das câmeras ajudam desenvolver a coordenação motora, agilidade, criatividade, percepção corporal e o equilíbrio físico e mental (Chamado de ponto de equilíbrio entre o cérebro e o corpo), a grande chave de tudo é que isso acontece de uma forma divertida, e muitas vezes o jogador nem percebe que desenvolveu estas habilidades. Angry Robots foi feito pensando nestes aspectos, pois o mesmo faz com que o usuário se movimente para poder atingir ou desviar de objetos e concluir com sucesso o objetivo do jogo.

REVISÃO DE LITERATURA

Não houve um jogo específico em que nos baseamos, queríamos algo diferente pouco utilizado, em questão do OpenCV e allegro juntos. Em busca de um jogo futurista, começamos pelo ambiente onde se passaria o enredo do mesmo, e para a nossa sorte a primeira imagem que apareceu era de um Cyborg, claro, a partir daquele instante sabíamos que a história teria um robô, como personagem principal. A segunda parte de escolhas foi adaptar a nossa ideia a realidade atual, precisávamos ambientar algo futurista em 2D, já que o allegro não nos permite ir muito além. Pesquisamos diversos jogos com robôs e um site em específico nos chamou atenção: Laboratório Garagem *LINK*, porém todos fugiam do ambiente proposto para nós, a maioria dos mesmo utilizavam arduinos, OpenCV e python, enquanto deveríamos desenvolver em Allegro 5, Linguagem C e OpenCV, a grande dificuldade foi o código de rastreamento já que não havia base.

DESENVOLVIMENTO

1. RGB

A conversão para escala de cinza é feita usando o sistema de cores RGB[5], (Red, Blue, Green).

Isso possibilita a utilização de todas as outras apenas usando azul, verde e vermelho, assim como o exemplo abaixo:



Figura 1: representação rgb

```
ALLEGRO_COLOR blue = al_map_rgb(0,0,255);
ALLEGRO_COLOR green = al_map_rgb(0,255,0);
ALLEGRO_COLOR red = al_map_rgb(255,0,0);
```

Este método permite um melhor rastreamento, mais rápido por não se tratar de 3 variáveis sendo usadas simultaneamente mas apenas com uma, onde seus valores variavam de 0 a 255. O valor 255 é o limite que a variável atinge por se tratar de uma única escala com base na captação de luz por uma câmera, onde cada bit tem valor máximo de 2⁸.

Essa limitação transforma as opções de tratamento em algo limitado, facilitando o número de algoritmos que podem ser usados

2. Mediana

A mediana[6] ($m=fi/2$) mostra o número central de um grupo, ou seja, os números são ordenados e o elemento que estiver no meio é a mediana, esse método é utilizado para tornar mais definido o degradê da luz, definindo assim uma borda onde as funções feitas para usar essa borda para fazer os próximos cálculos necessários para o programa obter a posição do objeto. Observe o exemplo abaixo:

```
int aux = 0;
unsigned char buffer[9];
for(int dy = -1; dy < 1; dy++){
    for(int dx = -1; dx < 1; dx++, aux++){
        buffer[aux] = imagem[y+dy][x+dx];
    }
}
for(aux = 0; aux < 8; aux++){
    for(int l = 0; l < aux; l++){
        if(buffer[l] > buffer[l+1]){
            int n = buffer[l];
            buffer[l] = buffer[l+1];
            buffer[l+1] = n;
        }
    }
}
imagem[y][x] = buffer[4];
}
else{
    return;
}
```

3. Limiar

O valor limiar define um limite mínimo de uma duração. Se uma duração ultrapassar do limite especificado o valor limiar será utilizado.

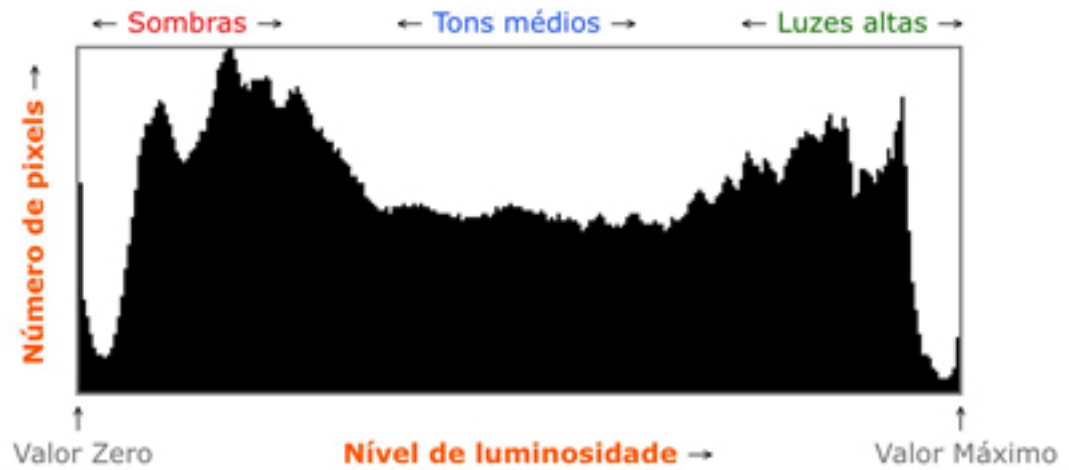


Figura 2: Exemplo de histograma

No jogo, o valor limiar é utilizado para o efeito de luz sobre o corpo, se o efeito da luz do ambiente passar do limite mínimo para a camera reconhecer o movimento ou objeto, o valor limiar irá ignorar a luz vindo do ponto específico do ambiente para que não atrapalhe no reconhecimento do movimento do jogo. Esse tratamento é conhecido como histograma, onde todos os valores são dispostos em um gráfico e todos os valores que superam a marca mínima são elevados ao valor máximo do histograma e os que não chegam a marca são rebaixados ao valor mínimo do gráfico.

Veja no exemplo abaixo:

4. Segmentação e Corte

O algoritmo de segmentação e corte foi utilizado para que se possa diminuir a zona de rastreamento do jogo, segundo pesquisas, os usuários não utilizam os cantos superiores e inferiores quando jogam, procuram se focar no centro da tela e consequentemente no centro da camera, dessa forma tendo uma melhor visão do jogo. Sabendo disso as partes exteriores da tela foram ignoradas, aumentando a velocidade do algoritmo e aprimorando a zona rastreada.

Veja o algoritmo que faz a segmentação e o corte:

```
int larg = largura / 8;
int alt = altura / 8;
int count = 1, count1 = 1;
while(count1 <= 8){
    int xo = 0;
    int x1 = larg;
    while(count <= 8){
        int yo = 0;
        int y1 = alt;
        if(xo != 0 && yo != 0){
            histograma(imagem, xo, x1, yo, y1);
        }
        yo += y1;
        y1 += alt;
        count++;
    }
    xo += x1;
    x1 += larg;
    count1++;
}
```

RESULTADOS

Na produção desse jogo foi utilizado conhecimentos amplamente desenvolvidos no 2º Semestre do curso com a biblioteca Allegro, o desenvolvimento principal trabalhado nesse projeto foi o da biblioteca OpenCV utilizando as funções limitadas pelo orientador o desenvolvimento foi em sua maior parte focado em algoritmos de alto desempenho focando a precisão e a variação de luminosidade, trabalhando de uma forma que o ajuste seja fácil.

CONSIDERAÇÕES FINAIS

Neste programa, o rastreamento pode ser considerado uma das partes mais demoradas e difíceis de desenvolver, pois alguns fatores atrapalharam, o principal deles foi a iluminação, dependendo do local o jogo não conseguia fazer o rastreamento correto, porém isso foi corrigido mais tarde usando cálculo de mediana e histograma ajustando-os de acordo com a iluminação. Foi constatado também que a conversão da imagem para uma única escala 256 melhora o processamento necessário e o gasto da memória. Em um programa que envolve um alto número de interações em um curto período de tempo é uma constatação essencial. O projeto proporcionou uma maior habilidade em lógica, linguagem C, cálculos com baixo custo computacional resultando em visão computacional, pois todos esses itens foram fundamentais para o desenvolvimento do jogo.

REFERÊNCIAS

- [1] Como Programar em C
H.M. Deitel P.J. Deitel
- [2] <http://www.rafaeltolledo.net/tutoriais-allegro-5/>
Allegro 5
- [3] <http://opencv.org/>
OpenCV
- [4] <http://noticias.r7.com/tecnologia-e-ciencia/noticias/jovens-apelam-para-games-corporais-para-melhorar-condicionamento-fisico-20110527.html>
Portal R7 de notícias
- [5] <http://www.significados.com.br/rgb/>
RGB
- [6] <http://educacao.uol.com.br/matematica/estatistica-moda-mediana.jhtm>
Cálculo de mediana
- [7] <http://www.ipb.pt/lmesquita/nova/04-05/mec/cap6.PDF>
Centróide

