

Interface Caligráfica de Escrita no Ar

Pedro Leitão

Dep. Eng. Informática, ISEP-Instituto
Politécnico do Porto

R. Dr. António Bernardino de Almeida,
431 Porto

pedromi-

guel_rs70@hotmail.com

António Castro

GILT/ISEP-Instituto Politécnico do
Porto

R. Dr. António Bernardino de Almeida,
431 Porto

avc@isep.ipp.pt

João J. Pereira

GECAD/ISEP-Instituto Politécnico
do Porto

R. Dr. António Bernardino de Almeida,
431 Porto

jjp@isep.ipp.pt

Resumo

Na actualidade, está a emergir um novo paradigma de interacção, designado por Natural User Interface (NUI), para reconhecimento de gestos produzidos com o corpo do utilizador. O dispositivo de interacção Microsoft Kinect, inicialmente concebido para controlo de videojogos, demonstra ser uma aposta viável para explorar outras áreas, como a do apoio ao processo de ensino e de aprendizagem para crianças do ensino básico. O protótipo desenvolvido visa definir um modo de interacção baseado no desenho de letras no ar, e realizar a interpretação dos símbolos desenhados, usando os reconhecedores de padrões KDA, SVM e $\$N$.

Abstract

A new interaction paradigm, named Natural User Interface (NUI), for the recognition of gestures produced with the user's body is emerging. The interaction device Microsoft Kinect, which was originally designed to control video games, proves to be a viable wager to explore other areas such as support to the teaching and learning process for children of elementary school. The developed prototype aims to define a mode of interaction for drawing letters in the air, and perform the interpretation of the symbols by means of the pattern recognizers KDA, SVM and $\$N$.

Keywords

Classificadores estatísticos, NUI, Kinect, Nuvens de pontos, Calibração, OpenNI, NITE, XNA, OSC, $\$N$.

1. INTRODUÇÃO

A interacção com equipamentos electrónicos usando teclado, rato ou outro controlador manual, tem vindo a ser bastante vulgarizada. O dispositivo Microsoft Kinect pretende colmatar as limitações humano-máquina existentes, através de uma interacção baseada em gestos realizados com o corpo. Este dispositivo, inicialmente desenvolvido para a utilização em videojogos, tem vindo a despertar interesse junto do público e a demonstrar diversas potencialidades. A aplicabilidade do Kinect não se limita unicamente aos videojogos, existindo um vasto leque de áreas nas quais pode ser utilizado. O âmbito deste projecto visa a aplicação do Kinect, juntamente com o reconhecimento de padrões, à área do ensino, integrando os algoritmos KDA, SVM e $\$N$ de forma a interpretar e identificar as letras correspondentes aos símbolos desenhados pelo utilizador, auxiliando assim o desenvolvimento motor e apoiando o processo de aprendizagem.

Nos tempos que correm, com a globalização e o crescimento tecnológico acelerado, as pessoas têm vindo a aderir cada vez mais a equipamentos digitais, em detrimento

da escrita convencional (papel e lápis). Contudo, e apesar de a sociedade se encontrar cada vez mais informatizada e dependente da tecnologia, a escrita convencional continua a ter um papel indispensável no quotidiano das pessoas, sobrepondo-se, em determinadas situações, à escrita digital. A escrita de textos, equações ou o desenho de esquemas nas salas de aulas constituem um bom exemplo da sua indispensabilidade. Estas tarefas tornam-se mais funcionais quando aliadas a uma interface tradicional.

Ao longo do tempo têm surgido diversas aplicações com este tipo de interfaces, refere Tomas Sylverberg [Sylverberg et al., 07]. Este autor propõe, como forma de apoio a militares, o uso de dispositivos móveis que lhes permitam desenhar símbolos NATO¹, essenciais na orientação dos soldados no terreno. O uso de dispositivos permitiria a partilha de informações do terreno, reportando a informação desenhada aos restantes elementos dispersos no campo.

¹ Conjunto de símbolos militares padronizados, para marcações em mapas pela NATO.

Nas secções 2 e 3 serão apresentados respectivamente alguns dos principais dispositivos NUI existentes no mercado e as correspondentes bibliotecas de desenvolvimento. A secção 4 aborda os algoritmos de reconhecimento de padrões analisados no âmbito deste projecto. O trabalho realizado e a respectiva avaliação são descritos nas secções 5 e 6, respectivamente. Por último, na secção 7, procede-se à conclusão do artigo e traçam-se as perspectivas de trabalho futuro.

2. HARDWARE

Com a mudança do paradigma de interacção, várias marcas de consolas de videojogos estão a lançar para o mercado novos produtos inovadores na área da interacção natural.

2.1 Wii

Nos anos 80, a Nintendo tentou, pela primeira vez, mudar o modo de interacção dos seus produtos com a criação da *Nintendo PowerPad*. Contudo, apenas anos mais tarde, em 2006, com o lançamento da *Wii*, o público aderiu consideravelmente a este modo de jogo. A *Wii* vem equipada com um sensor e um dispositivo semelhante a um comando de televisão, designado por *Wii Remote*, que utiliza um microcontrolador² ADXL330. Este controla três acelerómetros em simultâneo, um para cada eixo, de forma a obter a velocidade e a posição da mão do jogador [Tanaka et al., 12].

2.2 PlayStation Move

A multinacional japonesa *Sony* lançou para o mercado, no ano de 2004, o dispositivo de captura visual baseado em gestos denominado *Eye Toy*. Este conseguiu alcançar alguma notoriedade, despertando o interesse do público por este novo modo de interacção, apesar da baixa capacidade de captura de informação bidimensional de que dispunha. A *PlayStation Move* inclui o *Move Eye*, que consiste numa câmara RGB com uma resolução máxima de 640 por 480 píxeis, a funcionar a uma frequência de 60 a 120 *frames* por segundo (FPS). Contém também um conjunto de microfones e um dispositivo, designado por *Motion Controller*, idêntico a um bastão com uma esfera luminosa. O reconhecimento dos movimentos realiza-se com base na obtenção das coordenadas tridimensionais da esfera luminosa pela câmara RGB [Tanaka et al., 12].

2.3 Microsoft Kinect

O Kinect surgiu de uma conferência realizada em Junho de 2009 (Figura 1), sendo que o seu lançamento para o mercado aconteceu apenas em Novembro de 2010 [Tscherrig, 11]. O projecto que esteve na origem do Microsoft Kinect designou-se por 'Projecto Natal' e teve como base a tecnologia PrimeSense. Adoptava um sistema de luz padronizada e o microcontrolador PS1080 SoC, concebendo assim um dispositivo de controlo de videojogos para a consola Xbox 360 da Microsoft. Na sua constituição podemos encontrar um conjunto de microfones para aplicações que envolvam o reconheci-

mento de comandos por voz, uma câmara RGB com uma frequência de amostragem de 30 FPS e uma resolução de 640x480 píxeis. Além desta, inclui uma câmara de infravermelhos (IR) com uma resolução de 640x480 píxeis e frequência de amostragem de 30 FPS, usada para medir a profundidade dos elementos inseridos no cenário através da luz estruturada projectada sob eles e posteriormente reflectida para a câmara IR.



Figura 1 - Constituição do Microsoft Kinect [Mehlmann et al., 11]

3. SOFTWARE

A comunicação e acesso às funções do dispositivo Kinect processam-se através de um *driver*³ de comunicação entre o dispositivo e o sistema operativo. Os vários *drivers* de comunicação existentes, garantem o acesso a diferentes funcionalidades e suporte para diferentes sistemas operativos.

3.1 Microsoft Kinect SDK

A 21 de Maio de 2012 foi disponibilizada a versão comercial 1.5 da SDK Kinect. Esta versão contém um conjunto de ferramentas pensado para programadores e empresas, permitindo-lhes desenvolverem aplicações para este dispositivo. Esta versão comercializável veio permitir tanto a empresas como ao público em geral usufruírem de um conjunto de ferramentas para Windows direccionadas para o dispositivo Kinect [Pinto, 12b].

3.2 OpenNI

A plataforma OpenNI é uma plataforma para dispositivos NUI, que surgiu em Dezembro de 2010, fruto da associação de empresas sem fins lucrativos como a Willow Garage, desenvolvedora de hardware e software livre para aplicações pessoais robóticas [Willow Garage, 11], a Side-Kick, líder no desenvolvimento de jogos e publicação especializada de videojogos baseados em NUI [SideKick, 2011], e a fabricante de equipamentos informáticos ASUS, que comercializa o dispositivo NUI WaviXtion, baseado na tecnologia PrimeSense [Rogers, 11]. Através de um esforço conjunto estabeleceram uma plataforma padrão para equipamentos NUI, assegurando a compatibilidade, interoperabilidade e a comunicação entre estes equipamentos [OpenNI, 11a].

A sua infra-estrutura é segmentada por camadas (Figura 2), nas quais disponibiliza a interface para módulos de

² Consiste num computador num *chip*, contendo um processador, memória e periféricos de entrada/saída.

³ Pequeno programa que faz a comunicação entre o sistema operativo e o hardware.

dispositivos físicos e módulos para *middleware* de comunicação, associando-os à plataforma e acabando assim com problemas de incompatibilidades entre *middleware* e sensores ou vice-versa. Com isto, passa a ser possível desenvolver aplicações e portá-las para funcionar com outro *middleware* ou dispositivo sem que sejam necessárias modificações suplementares.

As camadas que constituem a infra-estrutura OpenNI são as que a seguir se discriminam [OpenNI, 11b]:

1. A camada aplicação funciona com base nos dados fornecidos da camada inferior e permite interagir com aplicações NUI alto nível;
2. Camada que representa as interfaces OpenNI, e os vários *middlewares* que podem ser associados à plataforma. Tem como objectivo processar os dados adquiridos pelos dispositivos e traduzi-los para serem interpretados pela camada mais acima;
3. Camada responsável por fazer a associação à plataforma dos diferentes dispositivos de captura de informação visual e áudio.

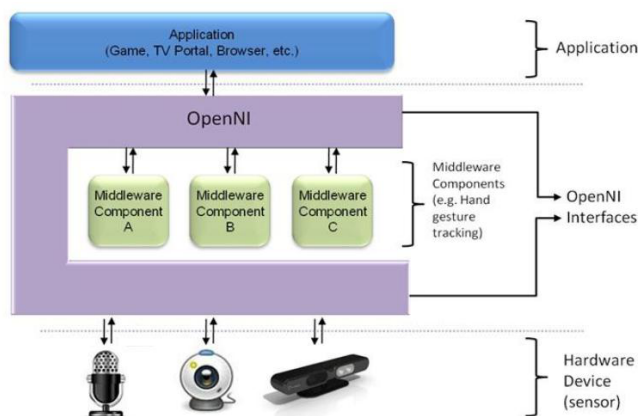


Figura 2 - Estrutura de camadas OpenNI [OpenNI, 11b]

4. ALGORITMOS DE RECONHECIMENTO CALIGRÁFICO

O reconhecimento de caracteres insere-se na área do reconhecimento de padrões, analisando características de uma imagem e decompondo-a de modo a extrair-lhe todas as particularidades que a caracterizam relativamente a outras amostras [Maxwell, 12]. Os reconhecedores dividem-se em dois grupos: estatísticos e conexionistas. Os classificadores estatísticos, nos quais se incluem os algoritmos KDA, SVM, \$1 e \$N, recorrem a um conjunto de amostras para aprendizagem antes do processo de classificação. Os classificadores conexionistas englobam algoritmos de redes neuronais e tentam simular as intercomunicações existentes entre os neurónios existentes num sistema neuronal biológico [Kolman and Margaliot, 09].

4.1 Kernel Discriminant Analysis

O KDA é uma extensão do algoritmo linear estatístico Linear Discriminant Analysis (LDA), usado no reconhe-

cimento de padrões e aprendizagem supervisionada [Yongmin et al., 01]. O KDA transpõe um problema para um espaço dimensional, representando as amostras através de vectores no espaço de características. Posteriormente, usando uma técnica Kernel, maximiza e determina não-linearmente a combinação de características representadas pelos vectores organizados em classes.

4.2 Support Vector Machine

O Support Vector Machine (SVM) é um algoritmo linear estatístico, descoberto por Vladimir Vapnik em 1963, aplicado no reconhecimento de padrões e aprendizagem supervisionada [Souza, 10b]. Este algoritmo procede a uma classificação linear binária dos dados de entrada, separando-os em duas categorias. O modelo de dados definido pelo SVM mapeia o conjunto de características através de vectores no espaço de características, separando-os em duas categorias, permitindo assim distinguir binariamente cada característica com base na categoria correspondente.

4.3 \$1

O \$1 é um método estatístico não-supervisionado conciso e simples, com uma taxa assinalável de reconhecimento de gestos [Anthony and Wobbrock, 2007a]. Funciona com base em geometria simples e trigonometria, que o torna simples e funcional, fazendo com que possa ser implementado em sistemas de prototipagem rápida. Neste são definidos dois conjuntos: o conjunto de pontos desenhados no momento para reconhecimento, designado de conjunto candidato C ; e as várias amostras de pontos existentes T_i , que servem como modelos de comparação. O \$1 caracteriza-se pela sua total invariância⁴ no que respeita à orientação dos gestos, fazendo com que, por exemplo, os sinais ' $>$ ' e ' $<$ ', sejam considerados como sendo o mesmo símbolo no processo de reconhecimento, não obstante as suas diferentes orientações.

A comparação entre os dois conjuntos, C e T_i , aplica o cálculo da distância euclidiana para determinar a melhor aproximação dos conjuntos de pontos existentes em T_i e que oferecem melhor alinhamento com o conjunto candidato C desenhado. Assim, com base nesta aproximação, é obtido o valor da menor distância d_i entre os vários conjuntos testados e quantificados, usando um sistema de pontuações, sendo que o que obtiver maior pontuação será o conjunto seleccionado [Wobbrock et al., 07b].

Antes do processo de procura e alinhamento do conjunto C , o caminho de pontos definidos por este é sujeito a quatro operações. A primeira passa por tornar os pontos de C equidistantes, reamostrando os pontos segundo um processo iterativo, no qual o distanciamento dos pontos varia consoante esse incremento. No segundo passo, procura rodar C através do ponto centróide⁵ definido pelo caminho dos pontos e pelo ponto em que se inicia o caminho dos pontos, realizando uma rotação para o ângulo indica-

⁴ Insensibilidade à rotação.

⁵ Ponto no interior de uma forma geométrica que define o seu centro geométrico.

tivo (0°) com base nestes dois pontos. Além de se encontrar o ângulo indicativo do conjunto C , são também procurados os ângulos indicativos dos conjuntos T_i aquando do seu carregamento, permitindo assim aperfeiçoar a procura do conjunto que melhor se alinha usando o método de alinhamento GSS⁶.

O terceiro passo resume-se a redimensionar não uniformemente o caminho de pontos para um quadrado tido como de referência, com um comprimento dos lados *size* igual em todas as amostras de T_i . Por fim, realiza a translação do centróide de C para a origem $(x, y) = (0, 0)$. Estes passos têm como intuito normalizar todos os gestos desenhados de forma a otimizar C , facilitando a correspondência dos pontos e o alinhamento com T_i , usando o GSS, como se pode observar na Figura 3 [Wobbrock et al., 07b].

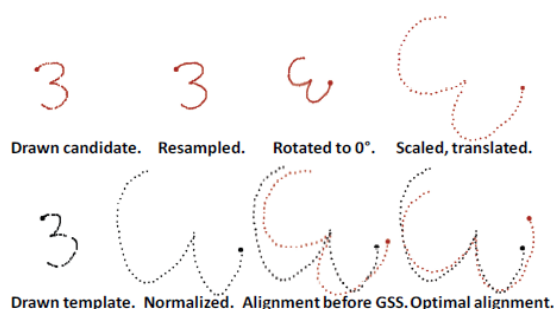


Figura 3 - Descrição dos passos de correspondência e alinhamento de um conjunto de pontos efectuado pelo \$1 [Anthony and Wobbrock, 07b]

4.4 \$N

O \$N baseia o seu funcionamento na mesma estratégia definida pelo seu predecessor \$1, realizando um conjunto de operações com base em geometria simples e trigonometria. Contudo, a grande melhoria encontra-se na versatilidade e capacidade de reconhecimento de gestos compostos por múltiplos traços [Anthony and Wobbrock, 07a].

Foram tidas em consideração duas questões, também abordadas por [Anthony and Wobbrock, 07b], que se prendem com a articulação dos gestos realizados e o efeito do número de amostras no rigor do reconhecimento. Relativamente à velocidade de articulação em que os gestos se processam, um aumento da velocidade não implica um aumento da taxa de erros no processo de reconhecimento, visto a distribuição dos pontos variar consoante o incremento do processo de reamostragem. Já o aumento do número de amostras melhora consideravelmente a precisão dos reconhecimentos.

O \$N foi sujeito a testes que permitiram determinar quais os melhores parâmetros a configurar, de forma a garantir uma taxa de reconhecimento razoável. As modificações

passaram pela alteração do número de pontos no processo de reamostragem em cada conjunto de pontos desenhados. A modificação do valor da reamostragem faz com que cada ponto seja representado com os espaçamentos entre pontos definido. Como resultado, ocorre um aumento ou diminuição da clareza dos pontos que constituem um traço. No gráfico da Figura 4 estão representados os 3 diferentes valores aplicados, 16, 64 e 96, num conjunto de 100 tentativas. Verifica-se que a reamostragem usando o incremento 64 oferece maior rigor.

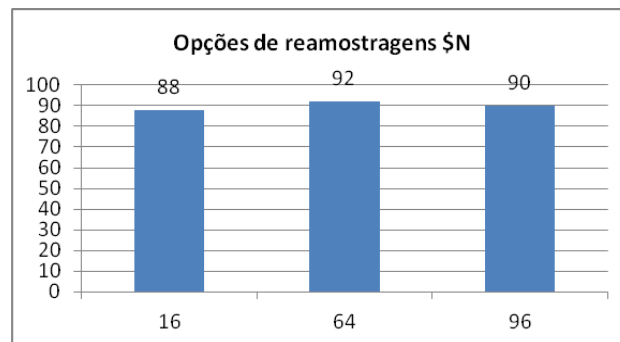


Figura 4 - Testes de eficiência do algoritmo \$N, usando diferentes valores de reamostragem

5. DESENVOLVIMENTO DA APLICAÇÃO

O software, desenvolvido em ambiente Windows, sobre a plataforma XNA Game Studio 3.1, visa definir uma interface de desenho de letras no ar que reconheça letras desenhadas por crianças do ensino básico. O dispositivo NUI Microsoft Kinect é usado como base para a concepção da aplicação, seguindo uma ordem de trabalhos que começa por localizar as articulações do corpo do utilizador, definir o plano de escrita e desenho das letras e, por fim, incorporar algoritmos de reconhecimento de padrões, para identificar as letras desenhadas.

Apesar de a versão oficial Microsoft Kinect SDK já se encontrar disponível actualmente, aquando do início deste projecto esta livreria estava apenas disponível na versão experimental, razão que levou a optar pelo uso da plataforma OpenNI. Outros factores influenciaram esta decisão, tais como o funcionamento com diferentes sistemas operativos, ser uma plataforma padronizada para qualquer equipamento, e tratar-se de uma plataforma de uso livre sujeita a constantes estudos, o que contribui para o aumento da sua maturidade e estabilidade.

Foram realizados testes com diferentes algoritmos de reconhecimento de padrões, que basearam a sua análise num conjunto de amostras representativas das 26 letras do alfabeto latino. Ambos os KDA e SVM usaram um conjunto de 260 amostras. Já o \$N usou um número bastante inferior, recorrendo para o efeito a apenas 41 amostras. A simplicidade e a flexibilidade na identificação de padrões, aliado ao reduzido número de amostras, fazem do \$N a melhor opção no processo de reconhecimento de padrões.

⁶ Técnica que consiste em encontrar o limite máximo ou mínimo recorrendo a uma função que analisa a distribuição dos valores dentro desses limites.

5.1 Interface com o utilizador

A aplicação desenvolvida pretende criar uma analogia com o modo de desenho de símbolos na areia da praia. Esta aplicação está dividida em seis grupos: detecção das articulações, configuração do painel de escrita, desenho de letras, reconhecimento de letras, gravação do desenho de letras e apresentação dos resultados das identificações.

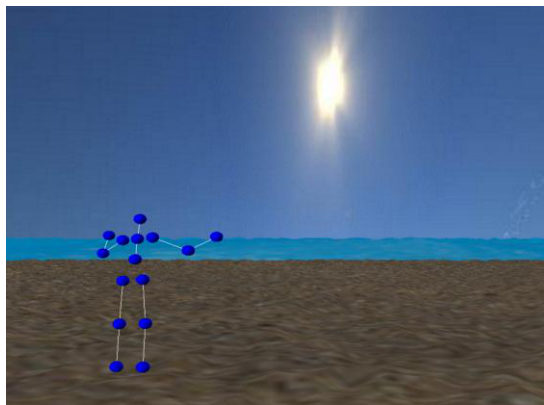


Figura 5 - Ecrã de localização das articulações do utilizador

O primeiro grupo, da detecção das articulações, destina-se à localização e rastreio do utilizador. Para isso, este deverá levantar os dois braços e baloiçá-los alternadamente para a esquerda e para a direita, até que o sistema o detecte e lhe reconheça as articulações do corpo. As articulações obtidas são posteriormente representadas através de esferas ligadas por intermédio de linhas, representando o esqueleto do utilizador e apresentando-o no cenário inicial da praia (Figura 5).



Figura 6 - Ecrã de indicação ao utilizador, destinado à selecção da mão de desenho que pretende usar

Depois de localizar o corpo, e ainda no cenário inicial da praia, segue-se a operação de Configuração do painel de escrita. Aqui, o utilizador deve estabelecer a distância entre si e o plano de escrita, movendo a mão direita para o lado direito, aproximadamente à mesma altura do ombro, e de seguida baixando a mão esquerda abaixo do nível da anca esquerda. Após definida a distância do uti-

lizador ao plano de escrita, aquele deve indicar qual das mãos pretende usar. Assim, quando aparece o ecrã ilustrado na Figura 6, deverá projectar para a frente a mão pretendida.



Figura 7 - Interface do ecrã de desenho

Depois de determinar os parâmetros inerentes à configuração do painel de desenho, é iniciada uma nova fase (Figura 7). Aí são concedidas cinco tentativas, nas quais o utilizador deverá acertar em cada uma das letras pedidas. Estas são geradas aleatoriamente de acordo com as 26 letras do alfabeto. O utilizador poderá movimentar o cursor (uma representação de um galho de uma árvore), movendo a mão de desenho ao longo de um plano situado mais próximo do seu corpo, ou então movimentar o cursor ao mesmo tempo que vai adicionando pontos ao ecrã, tendo a mão seleccionada de atravessar o plano mais afastado. Terminado o desenho estão disponíveis duas opções: a de limpar os pontos adicionados ou a de identificar o símbolo desenhado.



Figura 8 - Ecrã de selecção da letra a atribuir ao símbolo desenhado

Para identificar o conjunto de pontos que constituem o símbolo desenhado, surge um novo cenário para o grupo do reconhecimento de letras, onde é apresentado o resultado da identificação do símbolo, realizada pelo reconhecedor de padrões \$N. Aqui, estão unicamente disponíveis as opções para guardar ou não o referido símbolo. Caso não se pretenda guardar deve-se seleccionar a opção 'Não', sendo automaticamente direccionado para o grupo

de desenho de letras (o ecrã anterior, representado na Figura 7). Caso contrário, se seleccionar a opção para guardar o desenho, é então que surge um novo cenário (Figura 8), apresentando as 26 letras do alfabeto possíveis de associar ao conjunto de pontos do desenho que se pretende guardar. Para seleccionar a letra pretendida, o utilizador deverá projectar a mão de desenho até ao plano mais afastado, escolher a letra e, por fim, accionar o botão para guardar. Esta adição de um novo conjunto de pontos ao conjunto de amostras existentes enriquece o processo de reconhecimento e generalização de futuros desenhos.

Por fim, após terminarem as cinco tentativas disponíveis para o utilizador responder às letras geradas pela aplicação, são mostrados os resultados da identificação de cada símbolo pelo algoritmo de reconhecimento de padrões. As letras correctamente identificadas são assinaladas com um “visto” verde. Os erros são marcados com uma “cruz” vermelha.

5.2 Detalhes técnicos

De forma a obter as coordenadas das articulações, o utilizador deve começar por indicar a sua presença perante o dispositivo Microsoft Kinect (Figura 9), como referido anteriormente, até que as articulações sejam detectadas.



Figura 9 - Detecção da presença do utilizador, acenando com os braços acima da cintura

A definição do plano de escrita consiste, na realidade, num plano de dupla profundidade, estabelecido com base na medição da distância medida da mão direita ao ombro direito do utilizador (Figura 10).



Figura 10 - Postura definida para a determinação do duplo plano de interacção

As áreas de interacção no ar foram pensadas como um conjunto de três segmentos, definidos em função do valor obtido da medição referida anteriormente. Esse valor é segmentado em três (Figura 11), ficando assim o segmento que vai dos 0% aos 25% a tratar-se de um segmento de ócio. Assim, sempre que a mão de escrita se encontrar dentro deste segmento, não será desencadeada qualquer acção.

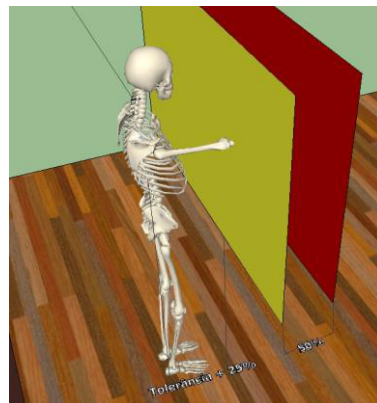


Figura 11 - Plano de escrita de duplo limite

O segmento seguinte é o maior dos três segmentos. Vai dos 25% aos 75% e é através deste que o utilizador deverá interagir maioritariamente, por se revelar essencial na movimentação e interacção com a aplicação. Sempre que o braço seja movimentado dentro destes limites, as coordenadas bidimensionais (x , y) da mão serão transpostas para o ecrã, movimentando o cursor também segundo as mesmas coordenadas. Neste segmento, os movimentos da mão do utilizador não se traduzem em desenhos no ecrã.

O último segmento vai dos 75% aos 100%. Nele, tal como no segmento anterior, é feita a transposição das coordenadas da mão para o cursor no ecrã. Neste caso, os gestos esboçados pelo utilizador são exibidos no ecrã.



Figura 12 - Utilizador dextro, com a mão projectada, prestes a ir de encontro ao limite estipulado para desenho

De seguida é necessário identificar qual das mãos será usada para desenhar as letras. Este processo foi pensado de forma a não limitar a interacção exclusivamente a utilizadores dextros, mas também ter em consideração o

público sinistro. Assim, o utilizador deverá projectar para a frente a mão que pretende usar de modo a interceptar o segmento de desenho (75%) como se encontra exemplificado na Figura 12.

Dos três algoritmos implementados, dois deles, o KDA e o SVM, recorrem a um método de classificação. Já o \$N\$ recorre a trigonometria simples, o que o torna mais rápido no processo de carregamento inicial das amostras em comparação com os dois anteriores. O \$N\$ apoia a sua análise num conjunto de 41 amostras, isto é, um número superior ao das 26 letras que constituem o alfabeto. Isto deve-se ao facto de cada uma das letras poder ser desenhada de diferentes formas, iniciando o traço de diferentes posições e variando o número de traços usados de acordo com a forma como a letra é traçada. Por exemplo, a letra 'A' pode ser desenhada com três traços ou simplesmente usando apenas dois.

Uma vez terminado um desenho, este é analisado pelo reconhecedor \$N\$, sendo indicado o resultado da identificação e apresentado o padrão que mais se assemelha de entre o conjunto de 41 amostras de aprendizagem usado.

6. VALIDAÇÕES

De modo a verificar a viabilidade da aplicação desenvolvida no âmbito deste projecto, procedeu-se a uma avaliação informal com um grupo de 20 participantes com idades compreendidas entre os 6 e os 25 anos, dos quais 7 elementos eram adultos, e os restantes crianças do ensino básico, tendo cada um deles realizado 10 tentativas. Ao longo dos testes foi possível verificar, por parte dos utilizadores, um certo embaraço nas duas tentativas iniciais, vindo progressivamente a adaptarem-se à interface ao longo das várias experiências realizadas. Considera-se positiva a adaptabilidade do utilizador à aplicação. No entanto, revelou-se essencial sujeitar o utilizador a um processo de aprendizagem. Ao longo das 10 tentativas, o utilizador vai-se familiarizando com os diferentes planos, adquirindo a prática necessária ao desenho das letras. Porém, após apenas três interações, o utilizador ainda demonstra debilidades relacionadas com a sensibilidade na transposição dos gestos, bem como dificuldades em situações em que se pretende remover a mão do desenho sem deixar rastros indesejados no ecrã. Estas situações carecem de um treino mais profundo para serem gradualmente corrigidas, melhorando consideravelmente a interação com o utilizador.

Relativamente ao reconhecedor de padrões eleito para a aplicação, o \$N\$, a definição de amostras com diferentes ordens de desenho das 26 letras do alfabeto revelou-se essencial para a precisão no reconhecimento, obtendo este algoritmo uma taxa de sucesso de 92%, em comparação com os outros reconhecedores de padrões Kernel Discriminant Analysis (KDA, 63%) e Support Vector Machine (SVM, 54%) testados (Figura 13).



Figura 13 - Grau de eficiência dos algoritmos de reconhecimento de padrões testados

7. CONCLUSÃO E TRABALHO FUTURO

O estudo deste sistema de reconhecimento de gestos no ar, tem vindo a permitir conceber uma solução direccionada às crianças do ensino básico de forma a auxiliá-las, na aprendizagem da escrita. Além disso, os desenhos produzidos são posteriormente analisados pela aplicação com uma taxa de reconhecimento bastante positiva, nas identificações efectuadas pelo reconhecedor. Até ao momento, com base nos testes realizados em dois grupos de pessoas de diferentes faixas etárias, foi possível constatar as valias desta aplicação na área da educação e desenvolvimento motor das crianças, assim como diferentes níveis da percepção espacial, por parte dos utilizadores das diferentes faixas etárias. Apesar de os resultados obtidos indicarem que o público-alvo mais jovem demonstra uma maior dificuldade ao longo do processo de familiarização com os planos, em comparação com os utilizadores de idade mais avançada. Este público mais jovem demonstra, em contrapartida, uma maior capacidade de adaptação, melhorando gradualmente as suas capacidades de coordenação.

Um aspecto essencial, a considerar para os utilizadores de qualquer faixa etária, prende-se com a necessária familiarização aos planos de interação, por forma a interagir eficientemente com a aplicação.

Como perspectivas de trabalho futuro, considera-se a utilização de um modelo dinâmico com um esqueleto interno, constituído por um conjunto de junções que façam correspondência com as junções captadas pelo Kinect/OpenNI, permitindo desta forma animar um modelo com os gestos realizados pelo utilizador.

8. REFERÊNCIAS

- [Anthony and Wobbrock, 07a] Anthony L., Wobbrock J. \$N\$ Multistroke Recognizer in JavaScript. *University of Maryland - Baltimore and University of Washington*, USA, 2007.
- [Anthony and Wobbrock, 07b] Anthony L., Wobbrock J. A Lightweight Multistroke Recognizer for User Interface Prototypes. *University of Maryland - Baltimore and University of Washington*, USA, 2007.

- [Kolman and Margaliot, 09] Kolman E., Margaliot M. Knowledge-Based Neurocomputing: A Fuzzy Logic Approach, Springer, p. 1-12, 2009
- [Maxwell, 12] Maxwell, Abril 2012.
<http://www.maxwell.lambda.ele.puc-rio.br/11498/11498_5.PDF>
- [Mehlmann et al., 11] Mehlmann G., Kistler F., Endrass B., André E., Klimmt C., Wagner J., Smirra R. Analysis of Interactive Storytelling Applications, UK, 2011.
- [OpenNI, 11a] OpenNI. Introducing OpenNI, Outubro 2011.
<http://75.98.78.94/images/stories/pdf/OpenNI_UserGuide_v4.pdf%20%5b1>
- [OpenNI, 11b] OpenNI. ProgrammerGuide, Outubro 2011.
< <http://openni.org/docs2/ProgrammerGuide.html>>
- [Pinto, 12b] Pinto P., 2012. SDK v1.5 do Kinect para Windows disponível para download, Maio 2012.
<<http://pplware.sapo.pt/windows/software/sdk-v1-5-do-kinect-para-windows-disponivel-para-download/>>
- [Rogers, 11] Rogers R., 2011. Journal. Kinect with Linux. Houston.
- [SideKick, 11] SideKick, , Outubro 2011.
< <http://www.sidekick.co.il/>>
- [Sousa, 12a] Sousa C., 2010. - Kernel Support Vector Machines for Classification and Regression in C#, Abril 2012.
<<http://crsouza.blogspot.pt/2010/04/kernel-support-vector-machines-for.html>>
- [Souza, 10b] Souza C., 2010. Handwriting Recognition Revisited: Kernel Support Vector Machines. The *Code Project*. Brazil.
- [Sylverberg T., 07] Sylverberg T., Kristensson P., Leifler O., Berglund E., Drawing on Paper Maps: Reliable On-line Symbol Recognition of Handwritten Symbols Using a Digital Pen and a Mobile Phone, 2007.
- [Tanaka et al., 12] Tanaka K., Parker J., Baradoy G., Sheehan D., Holash J, Katz L. A Comparison of Exergaming Interfaces for Use in Rehabilitation Programs and Research, Vol 6, 2012.
- [Tscherrig, 11] Tscherrig J., 2011. Activity Recognition using Kinect.
- [Willow Garage, 11] Willow Garage, Outubro 2011.
<<http://www.willowgarage.com/blog?page=8>>
- [Wobbrock et al., 07a] Wobbrock J., Wilson A., Li Y. \$1 Unistroke Recognizer in JavaScript. University of Washington and Microsoft Research, USA, 2007.
- [Wobbrock et al., 07b] Wobbrock J., Wilson A., Li Y. Gestures without Libraries, Toolkits or Training: A \$1 Recognizer for User Interface Prototypes. University of Washington and Microsoft Research, USA, 2007.
- [Yongmin et al., 01] Yongmin L., Shaogang G., Liddell H., Kernel Discriminant Analysis. University of London, *Department of Computer Science*, UK, 2001