

TRUE: um sistema para rastreamento, localização e identificação de usuários em ambientes inteligentes

Tales M. A. Porto¹, Danilo Á. M. C. Ferreira¹,
Fabricio N. Buzeto¹, Carla D. Castanho¹, Ricardo P. Jacobi¹

¹ Departamento de Computação – Universidade Federal de Brasília (UnB)

Abstract.

Resumo.

1. Introdução

A Computação Ubíqua, ou *ubiquitous computing*, vem sendo tema de diversas pesquisas desde o início dos anos 90. Mark Weiser [Weiser 1991, Weiser 1993] propôs que a computação devia ser algo invisível, nos envolvendo, servindo e exigindo o mínimo de esforço possível permitindo, ao usuário, ter mais foco na tarefa do que na ferramenta. Nesse cenário se destaca os ambientes inteligentes. Tais ambientes, também chamados de *smart spaces*, interagem com seus usuários auxiliando-os de modo proativo e transparente.

Ambientes são compostos por usuários, dispositivos e aplicações. Essas aplicações são softwares que provêm serviços de modo a facilitar as tarefas dos usuários no ambiente. Neste cenário, camadas de softwares denominadas de middlewares são responsáveis por orquestrar a interação e troca de informações entre as aplicações, os usuários e os dispositivos. Essas informações denominadas de informações de contexto são importantes na hora de prover inteligência ao ambiente.

A inteligência presente em um *smart space* é fruto de aplicações que levam em consideração as informações de contexto do ambiente, isto é, informações sobre os usuários e dispositivos.

São exemplos de informações de contexto:

- Temperatura do ambiente;
- Luminosidade;
- Informações dos usuários: número de usuários, frequência de entrada e saída, quem são eles, onde estão e o que estão fazendo;

Sem tais informações as aplicações não tem inteligência suficiente para interagir com os usuários de forma proativa. Essas informações são disponibilizadas pelo middleware as aplicações.

Dentre os inúmeros dados de contexto que podem ser extraídos do *smart space* destacam-se a localização e identidade do usuário, que são fundamentais para a implementação de serviços proativos e personalizados. Saber se existe algum usuário presente, quem é este usuário e onde ele está no momento pode ser fundamental para a implementação de aplicações inteligentes.

A obtenção de informações acerca da localização e identificação dos usuários torna-se um desafio tendo em vista a dinamicidade do ambiente, onde usuários entram e saem a todo momento, além de interagirem entre si e com diversos dispositivos.

Com a utilização de sensores é possível obter dados sobre os usuários, que podem ser utilizados em sistemas de rastreamento, localização e identificação, provendo assim, ao ambiente, importantes informações de contexto.

O objetivo desse trabalho foi o desenvolvimento de um sistema de rastreamento, localização, identificação de usuários em um ambiente inteligente denominado TRUE. TRUE é um acrônimo e vem de *Tracking and Recognizing User in the Environment*. O Sistema TRUE utiliza o sensor *Kinect* [Kinect 2012] da Microsoft como dispositivo de entrada. Esse dispositivo fornece imagens de cor e profundidade do ambiente. No caso específico, o foco do nosso trabalho foi prover essas informações ao Middleware *uOS* [Buzeto 2010].

O Middleware *uOS* é um projeto do grupo de pesquisa UnBiquitous da Universidade de Brasília que já existe desde 2007 coordenado pela Professora Carla Castanho e Professor Ricardo Jacobi cujo foco está na adaptabilidade de serviços em um ambiente de computação ubíqua, de modo que os serviços dos dispositivos presentes no ambiente possam ser oferecidos e compartilhados.

1.1. Trabalhos Correlatos

Agora serão mostrados alguns projetos que focam rastreamento, identificação e localização de pessoas em um ambiente inteligente. Alguns dos projetos estudados utilizam abordagens multimodais, abordagens que usam mais de um tipo de dado de entrada como dados audiovisuais.

1.1.1. Projeto CHIL

O Projeto CHIL (*Computers in the Human Interaction Loop*) [Stiefelhagen et al. 2008, Waibel et al. 2010] envolve uma rede de quinze laboratórios internacionais de pesquisa acadêmica e industrial.

O projeto foca no desenvolvimento de sistemas que auxiliam as interações entre grupos de pessoas como, por exemplo, sistemas que facilitam a colaboração em reuniões e em salas de palestras.

Dentre os protótipos que foram desenvolvidos no projeto destacam-se um ambiente de trabalho perceptivo e colaborativo e um sistema perceptivo de apoio a um escritório virtual.

Este projeto utiliza informações obtidas de diferentes algoritmos monomodais para criar um sistema de rastreamento e identificação multimodal a partir de dados audiovisuais das pessoas no ambiente.

O projeto utiliza além de dados visuais, uma rede distribuída de microfones para realizar o rastreamento de múltiplos alvos. A identificação é feita com imagens das faces providas a cada 200ms pelas várias câmeras no ambiente.

1.1.2. Smart Flow

Implementação de um sistema multimodal [Salah et al. 2008] que realiza detecção de movimento, rastreamento de pessoas, reconhecimento facial, localização baseada em áudio no middleware *SmartFlow*.

Ele consiste em um middleware genérico cliente-servidor que dá suporte para o processamento dos streams de dados e permite transporte de grande quantidade de dados de sensores para algoritmos de reconhecimento em nós distribuídos na rede.

A identificação e o rastreamento é feito por meio de imagens e a localização por meio de áudio.

O objetivo do sistema é identificar cada usuário ao entrar pela porta e rastreá-lo no ambiente. Para identificar os usuários quando entram o ambiente possui uma câmera voltada para a porta que fornece imagens de melhor qualidade.

1.1.3. AVIARY E MICASA

Nesse trabalho existem 2 ambientes: o AVIARY e MICASA [Trivedi et al. 2005]. Estes ambientes inteligentes são separados fisicamente, porém conectados. O primeiro (AVIARY) foi projetado para ser uma pequena sala de conferências. O segundo (MICASA) foi projetado para ser uma pequena sala de aula.

Nesse trabalho foi desenvolvido uma nova arquitetura para ambientes inteligentes chamado DIVA. Esta nova arquitetura pode ser vista como uma rede inteligente de câmeras, que são controladas para prover diversas funcionalidades, entre elas a detecção de intrusos, rastreamento de várias pessoas, análise de postura, identificação de pessoas e análise do movimento.

O sistema proposto por este projeto monitora o ambiente em baixa resolução de forma contínua, detectando somente a presença e suas dimensões. Formas de aquisição de imagens mais detalhadas são ativadas quando um evento ou atividade de potencial interesse é detectado. Esses eventos são os focos de atenção do sistema.

2. Fundamentação Teórica

O rastreamento pode ser definido como o problema de estimar a trajetória de uma entidade (objeto ou pessoa) em um plano de imagem à medida em que se move na cena. Em um ambiente inteligente, o rastreamento de pessoas é uma das principais ferramentas para detectar novos usuários e serve como base para que novas informações possam ser coletadas [Yilmaz et al. 2006].

Basicamente, o processo de rastreamento pode ser dividido em duas etapas:

- Detecção do objeto;
- Rastreamento do objeto detectado.

Antes de falar como são feitos a detecção e o rastreamento de entidades, vamos mostrar como as entidades rastreadas são representadas. As entidades rastreadas devem ser representadas de alguma maneira. Geralmente, as representações são baseados em

suas formas, existindo uma forte relação entre a representação e o algoritmo de rastreamento escolhido. Dentre as representações conhecidas se destacam a por pontos, por formas geométricas primitivas, por silhueta e contorno, por modelos de formas articuladas e por modelos de esqueletos. A representação por silhueta e contorno é mais indicada para rastrear entidades complexas de forma não rígida. Ela é popular devido à sua simplicidade e é mais utilizada no rastreamento de pessoas.

Todo método de rastreamento requer um mecanismo de detecção de entidades que pode ser realizada a cada *frame* (quadro) obtido ou na primeira vez que a entidade a ser rastreada entra no campo de visão. Dentre os métodos de rastreamento se destacam o detector de pontos, a subtração de fundo e a segmentação. A subtração de fundo é um método popular para segmentação de movimento, especialmente nas situações em que o plano de fundo é relativamente estático. Ele detecta as regiões de movimento na imagem obtendo a diferença pixel a pixel entre o quadro corrente e o quadro referente ao plano de fundo.

Os métodos de rastreamento de entidades mais utilizados atualmente são o rastreamento de pontos, o rastreamento de núcleo e o rastreamento de silhuetas. O rastreamento de silhuetas é feito estimando a região da entidade a cada quadro a partir das silhuetas geradas nos quadros anteriores. Dado os modelos de entidades, silhuetas são rastreadas por qualquer forma de correspondência ou evolução de contorno.

As informações provenientes do rastreamento de uma entidade servem como base para que a localização da mesma possa ser obtida. Com essas informações é possível determinar os pontos da imagem que representam a entidade e calcular as distâncias dos mesmos em relação a câmera utilizada na captura das imagens.

Imagens de profundidade são imagens cujo valor em cada pixel é uma função da distância do ponto, correspondente na cena, do sensor. Estas imagens podem ser adquiridas diretamente utilizando sensores específicos. Imagens de profundidade são úteis devido a sua especificação explícita de valores de profundidade.

Alguns dos métodos para se obter imagens de profundidade mais conhecidos são o tempo de voo (TOF - Time of flight) [Jain et al. 1995, Rougier et al. 2011] e a luz estruturada [Rougier et al. 2011]. Luz estruturada consiste em uma imagem de profundidade. Uma imagem de profundidade não pode ser obtida utilizando somente um sensor de vídeo. Porém, adicionando uma textura artificial na cena, uma imagem de profundidade pode ser recuperada. Esse princípio consiste na projeção de pontos de luz infravermelhos na cena que são recuperados por uma câmera infravermelha que lê a textura. Trata-se de um método mais acessível que o TOF, porém é pouco eficiente para estimar a distância dos pontos nas bordas dos objetos e em posições muito longe do sensor.

Hoje em dia, várias técnicas de reconhecimento biométrico por meio da face, íris, voz, entre outras, vêm sendo estudadas e utilizadas em sistemas de reconhecimento automático. Dentre estas, o reconhecimento por meio da face se destaca pois sua aquisição é realizada de maneira fácil e não intrusiva, tornando-a ideal para ser utilizada em um ambiente inteligente.

Do ponto de vista geral, o reconhecimento facial continua sendo um desafio por causa de várias dificuldades como iluminação, ângulos e poses, expressões, maquiagem e extração da face do contexto ou do fundo. Basicamente, o processo de reconhecimento

facial pode ser dividido em duas tarefas principais, a detecção de faces em imagens e o reconhecimento das faces encontradas.

A detecção de faces é definida como o processo que determina a existência ou não de faces em uma imagem e uma vez encontrada alguma face, sua localização deve ser apontada através de um enquadramento ou através de suas coordenadas dentro da imagem. Atualmente, já existem diferentes métodos/técnicas de detecção de faces.

O método é um dos mais utilizados e possui uma alta acurácia permitindo a detecção de faces em tempo real. Este método pode ser utilizado para construir uma abordagem de detecção facial rápida e eficaz utilizando apenas imagens em tons de cinza distinguindo-se dos outros métodos. Apesar da simplicidade obtém altas taxas de detecção. O método é implementado pela biblioteca OpenCV [Bradski 2000] e é amplamente utilizado.

O *Viola-Jones* [Ma 2007, P. Viola 2001] utiliza um método de aprendizagem de máquina chamado AdaBoost. Este combina vários classificadores “fracos” para criar um classificador “forte”. Um classificador fraco é aquele que só obtém a resposta correta um pouco mais frequente que um “palpite aleatório”. A combinação desses classificadores “fracos”, onde cada um “empurra” a resposta final um pouco na direção certa, pode ser considerado como um classificador “forte”. O método *Viola-Jones* combina uma série de classificadores AdaBoost na forma de uma cadeia de filtros, que recebe o nome de “Classificadores em Cascata”. Cada filtro em si é um classificador AdaBoost com um número relativamente pequeno de classificadores “fracos”.

Durante a utilização, se alguma região de uma imagem falhar em passar em um desses filtros, esta é imediatamente classificada como “não face”. Quando uma região de uma imagem passa por um filtro, ela vai para o próximo filtro na cadeia. As regiões das imagens que passarem por todos os filtros na cadeia são classificadas como “faces”.

Na etapa de reconhecimento, as faces detectadas, serão comparadas com um banco de dados de faces conhecidas. O *Eigenfaces* [Hewitt 2007] trata-se de uma técnica bastante satisfatória quando utilizada sobre uma base de dados (faces) relativamente grande. Ela permite ao sistema inferir das imagens suas principais características e, partindo delas, realizar o reconhecimento das imagens utilizando um número bastante reduzido de cálculos.

Esta técnica aplica o PCA (*Principal Component Analysis*) [P. N. Belhumeur 1996] nas imagens do banco de faces gerando uma lista de *eigenfaces*. A partir desta lista é calculada uma *eigenface* média. Cada *eigenface* possui um valor associado (*eigenvalue*) que representa a distancia da *eigenface* media da respectiva *eigenface*.

3. Sistema TRUE

O Sistema TRUE é um sistema aberto de rastreamento, localização e identificação de pessoas em ambientes inteligentes gerenciados pelo middleware *uOS*.

É um sistema monomodal implementado em C++ que utiliza somente dados visuais, como imagens de cor e profundidade. As imagens de profundidade são utilizadas no rastreamento e localização dos usuários no ambiente, enquanto que as imagens de cor são

utilizadas no reconhecimento facial e no cadastro dos usuários. Para se obter esses dados ele utiliza o sensor *Kinect* da Microsoft como dispositivo de entrada.

O Sistema TRUE é dividido em quatro módulos principais:

- Módulo de Rastreamento: responsável pelo rastreamento e localização dos usuários no ambiente.
- Módulo de Reconhecimento: responsável por identificar os usuários rastreados.
- Módulo de Registro: responsável pelo cadastro de novos usuários e treino do sistema.
- Módulo de Integração: responsável pela integração e comunicação do sistema com o Middleware *uOS*.

Basicamente, o Módulo de Rastreamento obtém o dados do sensor *Kinect* que são utilizados nas tarefas de rastreamento e localização de cada usuário no ambiente. Quando necessário ele requisita ao Módulo de Reconhecimento a identificação de algum usuário repassando imagens do mesmo. Por sua vez, o Módulo de Reconhecimento acessa o banco de faces, contendo as faces das pessoas cadastradas no sistema, e identifica o usuário mais parecido retornando seu nome ao Módulo de Rastreamento. Este último, repassa as informações de cada usuário ao Middleware *uOS* através do Módulo de Integração. Quando for necessário cadastrar um novo usuário no Sistema TRUE, o Middleware acessa o Módulo de Registro, também através do Módulo de Integração, que insere as imagens da nova pessoa na base de faces e retreina o sistema.

3.1. Middleware *uOS*

O Middleware *uOS* possui foco na adaptabilidade de serviços em um ambiente de computação ubíqua, de modo que os serviços dos dispositivos presentes no ambiente possam ser oferecidos e compartilhados.

O *uOS* é uma implementação de um middleware para auxílio de aplicações em ambientes de computação ubíqua, cuja proposta é compatível com os conceitos apresentados pela DSOA (*Device Service Oriented Architecture*).

O *uOS* possui três camadas:

1. Camada de Rede: responsável por gerenciar as interfaces de rede do dispositivo;
2. Camada de Conectividade: responsável por gerenciar conexões entre dispositivo que se comunicam por diferentes tecnologias, como Bluetooth e Ethernet;
3. Camada de Adaptabilidade: responsável por gerenciar os serviços disponíveis e o acesso aos mesmos.

A integração do Sistema TRUE com o middleware *uOS* foi realizada na Camada de Adaptabilidade através da implementação de um driver de “usuário”. Este driver se comunica com o Sistema TRUE disponibilizando serviços as aplicações para que as mesmas possam trocar informações. Esta integração será apresentada em detalhes mais a frente.

3.2. Módulo de Reconhecimento

O Módulo de Reconhecimento é responsável pela identificação dos usuários no ambiente utilizando a face como característica biométrica. A face foi escolhida por permitir um reconhecimento não intrusivo.

O reconhecimento facial é feito a partir de imagens de usuários que são repassadas pelo Módulo de Rastreamento. Estas imagens são compostas somente pela região em que o usuário se encontra.

Basicamente o reconhecimento facial possui as seguintes etapas:

1. Obtenção da imagem de entrada composta somente pelo usuário cujo reconhecimento foi requisitado.
2. Pré-processamento da imagem, ou seja, conversão da imagem em escala de cinza.
3. Detecção facial na imagem. Caso nenhuma face seja encontrada, retorna “vazio”. Observa-se que no máximo uma face pode ser encontrada nesta imagem.
4. Processamento da imagem, ou seja, uma nova imagem é criada recortando a região da face encontrada. A imagem é, então, redimensionada e equalizada criando assim um padrão de tamanho, brilho e contraste aumentando a acurácia do reconhecimento.
5. Reconhecimento facial com a técnica *Eigenfaces*.
6. Retorno do nome do usuário com a face “mais parecida” e a confiança do reconhecimento.

A detecção facial foi desenvolvida utilizando o método *Viola-Jones*. O *Viola-Jones* é implementado pela biblioteca *OpenCV* (*Open Source Computer Vision*). Basicamente, o processo de detecção facial procura por uma face em uma imagem pré-processada. Para realizar detecção facial utilizando o método *Viola-Jones* é necessário a utilização de um classificador em cascata. O Sistema TRUE utiliza um classificador treinado para detectar faces frontais em imagens. As etapas da detecção são:

1. Leitura de um classificador treinado para detectar faces em uma imagem.
2. Obtenção da imagem de entrada pré-processada.
3. Detecção da face na imagem utilizando o classificador.
4. Retorno da região da face detectada ou retorno “vazio” caso nenhuma face tenha sido encontrada. Observa-se que existe no máximo uma face na imagem, pois contém somente um usuário.

Foi desenvolvida utilizando a técnica *Eigenfaces*. Consiste em uma técnica bastante satisfatória quando utilizada sobre uma base de faces relativamente grande, permitindo ao sistema inferir, das imagens das faces suas principais características e, partindo delas, realizar o reconhecimento facial utilizando um número reduzido de cálculos, permitindo, assim, um reconhecimento em tempo real. A base de dados utilizada no Sistema TRUE é formada por imagens no formato PGM (*Portable Gray Map*) com tamanho de 92x112 pixels e em escala de cinza.

A base é composta por um banco de faces de alunos voluntários de Ciência da Computação da Universidade de Brasília. Um banco de imagens de faces da Universidade de Cambridge: formado por imagens de faces de 40 pessoas diferentes. Para cada pessoa, existem 10 diferentes imagens obtidas em diferentes momentos, com diferentes condições de iluminação, diferentes expressões faciais (olhos abertos e fechados, sorrindo e não sorrindo, entre outros) e diferentes detalhes faciais (com e sem óculos).

O reconhecimento facial inicia-se projetando a imagem no subespaço através do método PCA que reduz sua dimensionalidade. Então, calcula-se a distância da imagem

projetada a cada um dos *eigenfaces* obtidos na etapa de treinamento obtendo uma lista de distâncias. Esta lista de distâncias é comparada com a lista de distâncias de cada usuário, também obtidas na etapa de treinamento, obtendo o usuário cuja lista de distâncias é a mais similar.

Uma das etapas intermediárias consiste no cálculo da distância entre a imagem de entrada e os *Eigenfaces*.

Inicialmente, o cálculo desta distância era feito utilizando distância Euclidiana. Contudo, testes foram realizados com alguns usuários, em que o sistema realizava 20 tentativas de reconhecimento de usuários, e os resultados não foram satisfatórios. Portanto, os mesmos testes foram realizados utilizando a distância Mahalanobis. Ainda assim, os resultados não foram satisfatórios. Então, os mesmos testes foram feitos utilizando as duas distâncias de maneira conjunta: uma imagem só é tida como reconhecida quando o resultado das duas distâncias apontarem para a mesma identidade. Com isso, houve uma melhora significativa dos resultados.

3.3. Módulo de Rastreamento

O Módulo de Rastreamento é responsável por rastrear os usuários no ambiente, determinar a sua localização física em relação ao sensor *Kinect* e gerenciar suas identidades.

Para realizar o rastreamento e localização dos usuários é utilizada a biblioteca *OpenNI* (*Open Natural Interaction*). Trata-se de um *framework* que define *APIs* para o desenvolvimento de aplicações de interação natural.

Utilizando as imagens de profundidade, a detecção e o rastreamento dos usuários são feitos utilizando subtração de fundo. A representação das pessoas detectadas e rastreadas são feitas utilizando suas silhuetas.

Utilizando imagens de profundidade é possível calcular as coordenadas (x,y,z) de um plano cartesiano de três dimensões em que o ponto (0,0,0) corresponde a posição do *Kinect*. Dessa forma, a posição de um usuário em relação ao sensor é estimada utilizando as coordenadas referente ao pixel que representa seu centro de massa geométrico. Sendo assim, ao fixar a posição do *Kinect* no ambiente, é possível estimar a localização de qualquer usuário rastreado em tempo real.

O processo de rastreamento começa inicializando o dispositivo de entrada, no caso o sensor *Kinect*, registrando as ações a serem tomadas quando eventos em relação aos usuários ocorram, como, por exemplo, evento de novo usuário detectado ou usuário perdido. Obtém-se então uma imagem de fundo da cena que será utilizada no processo de subtração de fundo. Para cada imagem obtida da cena, o processo de subtração de fundo é realizado atualizando os parâmetros de cada usuário, como sua posição na imagem e sua posição em relação ao sensor.

O Módulo de Rastreamento detém as informações sobre todos os usuários rastreados no ambiente e é responsável por requisitar identificação ao Módulo de Reconhecimento. Isto acontece sempre que um novo usuário for detectado ou quando for necessário reconhecer novamente um usuário já rastreado.

Quando um novo usuário for detectado, o Módulo de Rastreamento obtém 20 imagens de cor sucessivas deste usuário. Para cada imagem, ele cria uma nova imagem

composta somente pela região em que este usuário se encontra e a envia ao Módulo de Reconhecimento requisitando identificação ao mesmo. O Módulo de Reconhecimento, por sua vez, realiza o reconhecimento facial e retorna ao Módulo de Rastreamento o nome do usuário e a confiança obtida.

A cada 0.5 segundos, o Módulo de Reconhecimento verifica se chegou algum resultado de identificação. Caso tenha chegado, o Módulo de Rastreamento computa o resultado e decide qual identidade será atribuída ao respectivo usuário.

Ao invés de realizar o reconhecimento somente quando novos usuários são detectados, com o objetivo de melhorar a confiança do reconhecimento, o Sistema TRUE realiza continuamente a identificação dos usuários já reconhecidos. Essas tentativas de reconhecer novamente os usuários ocorrerão a cada 5 segundos seguindo as mesmas etapas de quando um novo usuário for detectado. A única etapa que se difere é a primeira, ou seja, ao invés de obter várias imagens de um mesmo usuário, é obtida uma imagem de cada usuário rastreado e a mesma é enviada ao Módulo de Reconhecimento.

Ao obter um resultado de reconhecimento para determinado usuário, o Módulo de Rastreamento deve computar qual identidade será atribuída ao mesmo. Para isso, este módulo mantém para cada usuário o número total de vezes que este já foi reconhecido e os diferentes nomes obtidos pelo Módulo de Reconhecimento bem como a confiança média para cada nome e o número de vezes que cada nome foi atribuído àquele usuário.

Com todos esses dados, a identidade do usuário é definida por meio da Fórmula 1 que nada mais é do que a média ponderada entre as confianças de cada nome, onde os pesos utilizados são compostos pelo número de vezes que o respectivo nome foi atribuído ao usuário.

$$M_p = \frac{N_1 * C_1 + N_2 * C_2 + \dots + N_n * C_n}{N_1 + N_2 + \dots + N_n} \quad (1)$$

3.4. Módulo de Registro

O Módulo de Registro é responsável por cadastrar novos usuários no sistema e treiná-lo para também reconhecer esse novo usuário. Este consiste das seguintes etapas:

1. Obtenção das imagens do novo usuário;
2. Processamento das imagens, isto é, as imagens são convertidas em escala de cinza, novas imagens são criadas recortando a região da face encontrada. Em seguida, as imagens são redimensionadas e equalizadas criando assim padrões de tamanho, brilho e contraste;
3. Armazenamento das imagens;
4. Treinamento do sistema para reconhecer esse usuário.

A última etapa do módulo de registro consiste no treinamento do sistema. O treinamento inicia-se liberando os atuais dados de treinamento. Então, um vetor de imagens é obtido lendo todas as imagens contidas no banco de faces. Através deste vetor, obtém-se a *eigenface* média, os *eigenfaces* e os *eigenvalues*. Para cada usuário cadastrado no Sistema TRUE, suas imagens são projetadas no subespaço através do método PCA que reduz suas dimensionalidades, e são calculadas suas distâncias em relação aos *eigenfaces* obtendo um vetor de distâncias. Os *eigenfaces*, os *eigenvalues*, a *eigenface* média e os vetores de distâncias são armazenados e podem ser utilizados pelo Módulo de Reconhecimento.

Após o treinamento, o Sistema TRUE é reiniciado para que o reconhecimento seja feito utilizando as novas imagens e informações obtidas com o treinamento.

3.5. Modulo de Integração

O Modulo de Integração é responsável pela integração do Sistema TRUE com o Middleware *uOS*. Através desta integração, os dados providos pelo Sistema TRUE são disponibilizados ao ambiente gerenciado pelo middleware *uOS* por meio de serviços.

Para integrá-los foi desenvolvido um driver para o Middleware se comunicar com o Sistema TRUE. Este driver foi nomeado de *UserDriver*.

Os métodos do *UserDriver* podem ser divididos basicamente em três grupos:

- Métodos de Inicialização: métodos responsáveis pela inicialização do Sistema TRUE e pelo registro dos serviços que o *driver* possui e os eventos que pode gerar;
- Métodos Nativos: métodos declarados no *driver* que permitem acessar alguns métodos implementados pelo Sistema TRUE. Tais métodos permitem que o *UserDriver* tenha controle sobre o sistema, podendo iniciá-lo, encerrá-lo, retreiná-lo e, até mesmo, cadastrar e remover novos usuários a qualquer momento;
- Serviços e Eventos: são os métodos que implementam os serviços que o *UserDriver* disponibiliza às aplicações presentes no ambiente e os eventos que pode gerar.

O *UserDriver* disponibiliza às aplicações registradas no middleware os seguintes serviços:

- Consultas as informações dos usuários no ambiente: através dessas consultas, as aplicações tem acesso aos nomes, emails, posições correntes e confiança do reconhecimento de todos os usuários presentes no ambiente;
- Cadastro: as aplicações podem cadastrar novos usuários fornecendo ao *UserDriver* o nome, o email e as imagens do novo usuário;
- Treino do sistema: após cadastrar novos usuários as aplicações podem retreinar o sistema para poder reconhecer o novo usuário cadastrado;
- Remoção: as aplicações podem remover usuários cadastrados fornecendo o email do usuário.

Além dos serviços, o *UserDriver* gera os seguintes eventos:

- Novo Usuário: evento gerado assim que um novo usuário foi detectado pelo Sistema TRUE.
- Usuário Perdido: evento gerado assim que um usuário deixou de ser rastreado pelo Sistema TRUE.
- Atualização dos dados do usuário: evento gerado a cada cinco segundos atualizando os dados de todos os usuários rastreados.

4. Testes

Com intuito de identificar a acurácia e as limitações do Sistema TRUE perante seus requisitos foram realizados uma série de testes funcionais. Grande parte dos testes foram realizados no LAICO (LABoratório de Sistemas Integrados e CONcorrentes),

- Rastreamento: testes funcionais realizados para avaliar a eficiência da detecção de novos usuários no ambiente.
- Localização: testes realizados para avaliar a precisão do sistema ao estimar a posição dos usuários rastreados no ambiente.
- Reconhecimento: testes realizados para avaliar a acurácia na identificação de usuários.

Além disso, com o objetivo de demonstrar a integração do Sistema TRUE com o middleware uOS foi desenvolvida uma pequena aplicação utilizando os serviços e os eventos providos pelo sistema.

4.1. Rastreamento

O rastreamento é fundamental para o funcionamento correto do Sistema TRUE uma vez que é responsável por rastrear os usuários no ambiente, determinar a sua localização física em relação ao sensor *Kinect* e gerenciar suas identidades.

Os primeiros testes realizados tiveram como objetivo verificar a eficiência da detecção de novos usuários no ambiente. Os testes foram feitos simulando a entrada de um usuário na cena e analisando o momento em que o mesmo era detectado. Em todos os testes o usuário era detectado antes mesmo de entrar na área de visão do sistema por completo.

Também foram realizados testes com o objetivo de verificar a oclusão no rastreamento. Foi testado o caso em que um usuário oculta propositalmente outro já rastreado. Nesta situação, caso o usuário rastreado continuasse oculto, o sistema o determinava como perdido. Por outro lado, nos casos de oclusão parcial, o sistema se mostrou robusto.

Nos testes foi simulada também a situação de oclusão momentânea, ou seja, um usuário em movimento oculta outro por um curto período de tempo em razão da sua movimentação. Nestes casos, o sistema TRUE perde o usuário rastreado, mas rapidamente é capaz de detectá-lo novamente.

A oclusão era um problema esperado uma vez que o Sistema TRUE utiliza somente um sensor *Kinect* como dispositivo de entrada.

Durante os testes realizados com rastreamento foi observado alguns problemas quando o usuário rastreado interagiu com objetos do ambiente ou com outros usuários. Na grande maioria das vezes em que o usuário interagiu com objetos, o Sistema TRUE considerou o objeto como sendo parte do usuário. Entretanto, esta situação não prejudicou a eficiência do sistema.

Por outro lado, os problemas com interação entre usuários foram mais raros, porém tiveram impacto maior. Tais problemas consistem em “interferências” que ocorreram em algumas situações de contato entre dois ou mais usuários.

Apesar dos problemas relatados, o rastreamento conseguiu, na maioria dos testes, atender às necessidades rastreando os diversos usuários no ambiente em suas atividades diárias.

4.2. Localização

O Sistema TRUE obtém a localização dos usuários no ambiente por meio de coordenadas dos mesmos em relação ao *Kinect*. Para verificar a acurácia dessas coordenadas foram

realizados alguns testes.

As coordenadas (x , y , z) obtidas pelo sistema são coordenadas de um plano cartesiano de três dimensões em que o ponto (0, 0, 0) corresponde a posição do *Kinect*. A coordenada no eixo z mostra o quanto o usuário está distante, em termos de profundidade, em relação ao sensor, a coordenada no eixo x mostra o quanto o usuário está a direita ou a esquerda do sensor e a coordenada no eixo y mostra o quanto o centro de massa geométrico do usuário está acima ou abaixo do sensor.

Os testes foram realizados para as coordenada do eixo x e z pois o eixo y se refere a altura do centro geométrico do usuário rastreado e não é interessante para o contexto de localizar o usuário no ambiente.

Para testar a precisão as coordenadas do eixo z foi realizado o seguinte teste: um objeto (uma caixa de papelão) foi colocada em frente ao sensor a diferentes distâncias do mesmo (1000mm, 2000mm, 3000mm, 4000mm, 4057mm). Para distâncias acima de 4057mm não foram obtidos resultados pois essa é a distância máxima de funcionamento do sensor.

Para cada distância, foram coletados 6 amostras informadas pelo sistema. Foi observado, a discrepância aumenta conforme o objeto se distancia do sensor. Neste teste o menor erro obtido foi de 3,21mm e o maior de 111,75mm.

Para testar a precisão das coordenadas foi realizado o seguinte teste: um objeto (uma caixa de papelão) foi colocada a uma distância fixa de 3000 milímetros do sensor e colocada em diferentes posições ao longo do eixo x (0, $\pm 330mm$, $\pm 660mm$, $\pm 990mm$). Para cada posição foram coletadas 6 amostras informadas pelo sistema. Foi observado que existe uma diferença constante de poucos centímetros ao longo de todo teste. É possível inferir então, que o Sistema TRUE consegue fazer estimativas das coordenadas no eixo x com erro de poucos centímetros. Neste caso o menor e o maior erro obtido foi de 27,19mm e 79,29mm respectivamente.

Acredita-se que os erros apresentados não são significativos a ponto de comprometer o uso dos dados fornecidos.

4.3. Identificação

Para análise dos resultados foi utilizada como ferramenta uma matriz de confusão construída a partir dos dados coletados do sistema. Tal matriz contém em suas colunas os valores possíveis que se podem obter para cada usuário e as linhas os usuários que foram observados.

Sumarizando os resultados expressos na matriz apresentamos três taxas calculadas:

1. Verdadeiro Positivo: quando o sistema identifica o usuário de maneira correta.
2. Verdadeiro Negativo: quando o sistema identifica o usuário de maneira errada.
3. Falso Negativo: quando o sistema não identifica o usuário cadastrado.

Os cenários de teste utilizados para se averiguar a precisão do reconhecimento facial consistiram de duas etapas. Na primeira etapa cada usuário foi cadastrado na base de dados utilizando estratégias distintas para cada cenário. A esta base de dados foram adicionadas os dados das 40 pessoas presentes no banco de faces da Universidade de

Cambridge. Na segunda etapa cada usuário, individualmente, foi posicionado dentro do campo de visão do sensor onde foram coletados os resultados dos primeiros 20 valores retornados pelo sistema. Primeiramente foi construído um cenário de testes que não apresentou resultados satisfatórios, com isto foi construído um segundo cenário de teste com uma nova abordagem para a coleta de faces (1ª etapa).

No primeiro conjunto de testes, os cadastros dos usuários foram feitos utilizando 10 imagens de faces de cada usuário: 6 imagens frontais da face, 2 imagens da face ligeiramente rotacionada para direita e 2 imagens da face ligeiramente rotacionada para esquerda. Os cadastros foram realizados com essa estratégia com o objetivo de diminuir o impacto das variações de poses e ângulo no reconhecimento facial. Os resultados obtidos dos processos de reconhecimento facial realizados para cada usuário são mostrados na matriz de confusão.

Como previsto, foi possível observar na matriz que os valores mais significativos estão na diagonal principal. Alguns resultados obtidos foram satisfatórios com porcentagens maiores que 90%. Por outro lado, foram obtidos alguns resultados com baixas porcentagens como 45%. Estes últimos foram principalmente causados por problemas como pose e expressões faciais. A iluminação não foi um problema, pois no ambiente de teste não existe influência da iluminação externa e a interna é controlada.

As taxas de Verdadeiro Positivo, Verdadeiro Negativo e Falso Negativo foram calculadas analisando os valores da matriz. Observa-se que a taxa de verdadeiro positivo (73,63%) foi uma taxa baixa para um sistema de reconhecimento automático. Neste conjunto de testes, a taxa de verdadeiro negativo foi bem superior a taxa de falso negativo mostrando que a principal deficiência do reconhecimento facial consiste na confusão entre os usuários cadastrados no sistema.

Os resultados deste primeiro cenário não se mostrou satisfatório. Sob a hipótese do resultado ter sido influenciado pela baixa quantidade de faces e poses, construiu-se um segundo cenário de testes. Neste a estratégia utilizada nos cadastros dos usuários foi diferente. Os cadastros foram feitos utilizando 100 imagens das faces de cada usuário em diferentes ângulos, posições e expressões faciais. No momento do cadastro, foi solicitado que os usuários movimentassem o rosto levemente para cima, para baixo, para esquerda e para direita de maneira aleatória e com diferentes expressões faciais.

Houve um aumento na taxa de Verdadeiro Positivo de 73,63% para 82,5% e redução significativa na taxa de Verdadeiro Negativo de 17,27% para 6,0%, mostrando que os casos de confusões entre os usuários foram mais raros. A melhora obtida comprova a hipótese que um número maior de poses e faces aprimorou a capacidade do sistema de reconhecer seus usuários. Este acréscimo de informações torna os resultados mais robustos devido a inclusão de informações de mais cenários dos usuários no sistema.

4.4. Integração

Com intuito de exemplificar a utilização do *UserDriver* e testar a integração do Sistema TRUE com o middleware *uOS* foi desenvolvido uma aplicação para o middleware chamada *UserApp*. Esta aplicação registra um *listener* para “escutar” os eventos do *UserDriver* chamado *UserListener*.

Quando o *UserListener* é inicializado ele obtém acesso ao Twitter e se registra

para escutar os eventos gerados pelo *UserDriver*. Para cada evento obtido, ele envia mensagens pelo Twitter para os usuários no ambiente. Para eventos de “novo usuário detectado” e “usuário perdido”, ele envia mensagens de boas vindas e de despedidas respectivamente. O *UserDriver* também gera eventos de atualização dos dados dos usuários. Quando estes eventos são gerados, o *UserListener* verifica se houve atualização na identidade do usuário e se o mesmo está a mais de uma hora no mesmo lugar. Caso esteja, ele envia mensagens aos usuários informando estes acontecimentos.

Testes funcionais foram feitos com a aplicação mostrando que o *driver* consegue obter os dados íntegros do Sistema TRUE e gerar os eventos de maneira praticamente instantânea. A Figura mostra as mensagens geradas pela aplicação em um teste funcional, onde Danilo, um usuário cadastrado, entra no ambiente senta em uma mesa com seu notebook permanecendo no mesmo lugar por mais de uma hora, e logo depois deixa o ambiente.



Figura 1. Exemplo das mensagens enviadas pelo Twitter aos usuários no ambiente.

5. Conclusão

Foram conduzidos conjuntos de testes para cada um dos propósitos do Sistema TRUE. Os testes de rastreamento simularam diferentes situações diárias onde os usuários rastreados interagem entre si e com diferentes objetos. O Sistema TRUE se mostrou eficaz tendo em vista que os novos usuários são detectados antes mesmo de entrarem por completo no campo de visão do sensor. Além disso, revelou-se robusto em casos de oclusões parciais e como esperado, nos casos de oclusões totais não foi possível prosseguir com o rastreamento. Os testes também mostraram que os limites do sensor permitem até 5 pessoas a uma distância de 4,057 metros.

Os testes de localização foram realizados comparando as coordenadas obtidas pelo Sistema TRUE com as coordenadas mensuradas manualmente. As coordenadas obtidas pelo sistema se mostraram consideravelmente próximas das reais, apresentando diferenças de poucos centímetros que aumentam juntamente com a distância entre o usuário e o sensor.

Foram realizados dois cenários de testes de identificação que se diferem na estratégia utilizada no cadastro dos usuários. O segundo cenário utiliza uma maior quantidade de imagens da face de cada usuário com uma maior variação dos ângulos, poses e

expressões faciais. Os testes mostraram que a estratégia utilizada no cadastro dos usuários tem grande impacto nos resultados do reconhecimento, sendo que a taxa de acertos aumentou de 73,63% (primeiro cenário) para 82,5% (segundo cenário).

Com as informações providas pelo Sistema TRUE, tornou-se possível o desenvolvimento de novas aplicações que possuem um maior entendimento do contexto do ambiente, como pode ser visto no *UserApp*. Esta aplicação utiliza as identidades e posições dos usuários para enviar mensagens de boas vindas, despedidas e mensagens para informá-los a quanto tempo estão no mesmo lugar.

5.1. Trabalhos Futuros

Os dados providos pelo Sistema TRUE e disponibilizados pelo Middleware uOS constroem uma nova base para a criação de inúmeras aplicações para ambientes inteligentes. A partir das informações de identidade e localização dos usuários, tornou-se possível desenvolver aplicações de reconfiguração automática de serviços, estabelecimento de contextos e definição de perfis.

Além disso, o Sistema TRUE pode ser melhorado em algumas de suas características. Uma dessas melhorias consiste na expansão do sistema para permitir a utilização de múltiplos Kinects de modo a expandir sua área de cobertura. Essa expansão minimizaria os problemas de oclusão e também aumentaria o número máximo de usuários rastreados.

É possível também melhorar os resultados do reconhecimento. Como visto nos testes, foi possível obter melhores resultados apenas modificando a estratégia de cadastro. Essa melhoria pode ser desenvolvida em duas frentes, como estudo de novas formas de cadastros de usuário e novas técnicas que utilizam os dados de profundidade para o reconhecimento facial.

Referências

- Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.
- Buzeto, F. N. (2010). Um conjunto de soluções para a construção de aplicativos de computação ubíqua. Master's thesis, Departamento de Ciência da Computação, Universidade de Brasília, <http://monografias.cic.unb.br/dspace/handle/123456789/257>.
- Hewitt, R. (2007). Face recognition with eigenface. *SERVO Magazine*.
- Jain, R., Kasturi, R., and Schunck, B. G. (1995). *Machine vision*. McGraw-Hill, Inc., New York, NY, USA.
- Kinect (2012). Kinect. <http://www.xbox.com/en-US/kinect>.
- Ma, E. L. H. (2007). Avaliação de características haar em um modelo de detecção de face. *Universidade de Brasília, Instituto de Ciências Exatas, Departamento de Ciência da Computação*.
- P. N. Belhumeur, J. P. Hespanha, D. J. K. (1996). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *European Conference on Computer Vision*.
- P. Viola, M. J. (2001). Robust real-time object detection. *Second International Workshop on Statistical and Computational Theories of Vision – Modeling, Learning, Computing, and Sampling*.

- Rougier, C., Auvinet, E., Rousseau, J., Mignotte, M., and Meunier, J. (2011). Fall detection from depth map video sequences. In Abdulrazak, B., Giroux, S., Bouchard, B., Pigot, H., and Mokhtari, M., editors, *ICOST*, volume 6719 of *Lecture Notes in Computer Science*, pages 121–128. Springer.
- Salah, A. A., Morros, R., Luque, J., Segura, C., Hernando, J., Ambekar, O., Schouten, B., and Pauwels, E. (2008). Multimodal identification and localization of users in a smart environment. *Journal on Multimodal User Interfaces*, 2(2):75–91.
- Stiefelhagen, R., Bernardin, K., Ekenel, H. K., and Voit, M. (2008). Tracking identities and attention in smart environments - contributions and progress in the chil project. In *FG*, pages 1–8.
- Trivedi, M. M., Huang, K. S., and Mikic, I. (2005). Dynamic context capture and distributed video arrays for intelligent spaces. *Ieee Transactions On Systems, Man, and Cybernetics—part A: Systems and Humans*, 35(1):145 – 163.
- Waibel, A., Stiefelhagen, R., Carlson, R., Casas, J. R., Kleindienst, J., Lamel, L., Lanz, O., Mostefa, D., Omologo, M., Pianesi, F., Polymenakos, L., Potamianos, G., Soldatos, J., Sutschet, G., and Terken, J. (2010). Computers in the human interaction loop. In *Handbook of Ambient Intelligence and Smart Environments*, pages 1071–1116.
- Weiser, M. (1991). The computer for the 21st century. *Scientific American*.
- Weiser, M. (1993). The world is not a desktop. *ACM Interactions*, 1.
- Yilmaz, A., Javed, O., and Shah, M. (2006). Object tracking: A survey. *ACM Comput. Surv.*, 38(4):13+.