

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 ubicomp, vem sendo tema de diversas pesquisas desde o início dos anos 90. Neste sentido temos que a computação devia ser algo invisível, nos servindo e exigindo o mínimo de esforço possível [Weiser 1991, Weiser 1993]. Permitindo assim, que o usuário tenha mais foco na tarefa em execução que na ferramenta. Um ambiente com tais características é chamado de inteligente (*smart space*) pois busca auxiliar seus usuários de modo proativo e transparente.

A inteligência esperada nestes ambientes é fruto de aplicações que tratam os dados fornecidos pelos diversos dispositivos presentes, bem como coordenam suas ações. Para que tudo isto ocorra de forma transparente estes softwares devem ser sensíveis ao contexto em que estão presentes. Tendo em vista a dinamicidade do *smart space*, uma vasta gama de informações podem ser utilizadas para se construir este contexto. De maneira especial podemos olhar para as informações de identificação e localização do usuário. De posse destas informações é possível personalizar as ações bem como direcioná-las aos locais corretos do ambiente.

O objetivo desse trabalho foi o desenvolvimento de um sistema de rastreamento, localização e identificação de usuários em um ambiente inteligente. Tal sistema foi denominado TRUE, um acrônimo para *Tracking and Recognizing User in the Environment*. O Sistema TRUE obtém informações do ambiente utilizando o sensor Kinect [Kinect 2012] da Microsoft e disponibiliza seus resultados para as aplicações através do Middleware uOS [Buzeto et al. 2011, Buzeto 2010].

2. Fundamentação Teórica

Em ambientes inteligentes, as informações de contexto sobre os usuários como localização e identidade são de grande importância proporcionando uma maior acurácia nas tomadas de decisões. Informações de contexto como essas apresentam diversos desafios na sua obtenção tendo em vista que os usuários se movimentam e interagem entre si e com os dispositivos de forma dinâmica.

Dentre as diversas formas de se identificar um usuário, como pela íris e padrão de voz, o reconhecimento por meio da face se destaca devido a sua aquisição ser realizada de maneira não intrusiva. Porém o reconhecimento desta informação apresenta desafios

devido a variações de iluminação, ângulos e poses, expressões, maquiagem, dentre outros. Neste cenário, o processo de reconhecimento facial pode ser dividido em duas tarefas principais, a detecção das faces nas imagens e o reconhecimento destas.

A detecção de faces é definida como o processo que determina a existência ou não de faces em uma imagem. Atualmente, existem diferentes métodos/técnicas de detecção de faces. Um dos mais utilizados é o *Viola-Jones* [Ma 2007, P. Viola 2001]. 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.

Para o reconhecimento, busca-se padrões nas faces de forma que estes possam ser usados para a identificação do usuário. A técnica *Eigenfaces* [Hewitt 2007] consiste em extrair as principais características de uma face e com base nelas é determinado um valor correspondente, o *eigenvalue*. Para tal é aplicada a técnica de PCA (*Principal Component Analysis*) [Belhumeur et al. 1996] nas imagens já conhecidas. Com isto se obtém um valor de *eigenvalue* médio bem como a distância de cada face para este valor. Assim quando se identifica uma nova face, se calcula o valor do *eigenvalue* e se compara com as distâncias conhecidas. Esta técnica apresenta bons resultados e é amplamente utilizada em bibliotecas como a OpenCV [Bradski 2000].

Para a localização do usuário podemos utilizar a técnica de imagens de profundidade onde a cada pixel da imagem é atrelado a um valor de distância relativa ao sensor. Dentre as técnicas conhecidas podemos destacar a TOF (*Time of Flight*) [Jain et al. 1995, Rougier et al. 2011] e luz estruturada [Rougier et al. 2011]. 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 pouco eficiente para estimar a distância nas bordas dos objetos e em posições muito longe do sensor.

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 e rastreamento do objeto detectado. Existe um forte relacionamento entre as técnicas de rastreamento e a representação usada pelos mesmos. Dentre as representações conhecidas destacam-se 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 a sua simplicidade sendo muito 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.

3. Trabalhos Correlatos

Rastreamento, localização e identificação de usuários é um tema que tem motivado diversos trabalhos na academia. Aqui apresentaremos alguns projetos que se destacam dentro desta área de pesquisa.

O Projeto CHIL [Stiefelhagen et al. 2008, Waibel et al. 2010] possui foco em auxiliar nas interações de pessoas em ambientes colaborativos, como salas de reuniões e palestras. Ele se utiliza de dados de áudio e vídeo do ambiente como fontes de informação. Porém, para a identificação e localização do usuários são utilizados apenas o áudio coletado através de um vetor de microfones.

O *SmartFlow* [Salah et al. 2008] é um middleware que realiza a detecção de movimento, rastreamento de pessoas, reconhecimento facial, localização baseada em 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.

O AVIARY e MICASA [Trivedi et al. 2005] são dois *smart spaces*. 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. 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.

Os projetos aqui apresentados mostram avanços na localização e identificação de usuários. Cada um busca utilizar os dados mais importantes ou abundantes encontrados para estas finalidades (nos casos citados: áudio ou vídeo). Independente da fonte de informação quando a característica de identificação é interrompida (a face do usuário não está voltada a câmera ou o usuário não está falando) não temos o rastreamento do mesmo. Sendo assim, manter esta informação relacionada a outros dados disponíveis permitiria rastrear o usuário mesmo durante momentos de oclusão de sua identificação. Este é o foco do sistema TRUE aqui apresentado.

4. Sistema TRUE

O Sistema TRUE (*Tracking and Recognizing Users in the Environment*) realiza a identificação, localização e rastreamento de usuário em um ambiente inteligente.

Para isso são utilizados os dados de vídeo e profundidade providos pelo sensor *Kinect* [Kinect 2012]. Por fim estas informações são disponibilizadas para as aplicações através do middleware *uOS* [Buzeto et al. 2011, Buzeto 2010, Gomes 2007]. O projeto é composto por 4 módulos conforme representado na Figura 1.

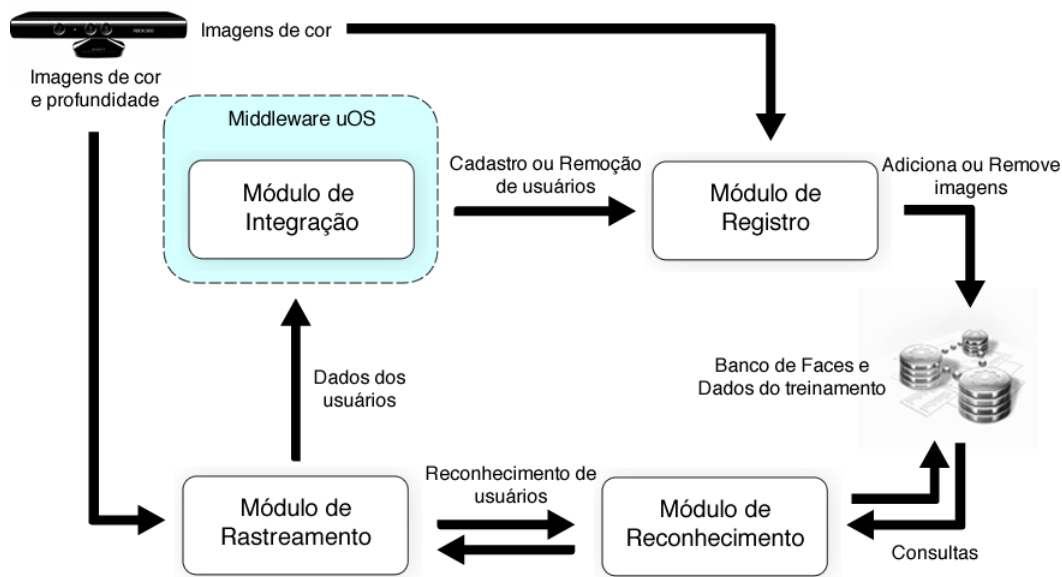


Figura 1. Interação entre o módulos do Sistema TRUE.

4.1. 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*) [OpenNI 2010].

Para a detecção e rastreamento do usuário é utilizada a técnica de subtração de fundo com a representação de silhuetas. A localização, por sua vez, é realizada utilizando-se imagens de profundidade que fornecem as coordenadas do centro de massa do usuário. Tais coordenadas são compostas por três dimensões onde o sensor é considerado como a origem do plano.

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. De cada imagem é extraída a região correspondente a face e enviada 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 500ms, 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. Após o reconhecimento inicial, o módulo de rastreamento realiza uma nova tentativa de reconhecimento a cada 5 segundos. Desta forma o sistema possibilita uma melhora no reconhecimento obtido.

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. Desta forma a identidade é definida pela média ponderada das confianças obtidas, escolhendo-se a de maior valor obtido.

4.2. Módulo de Reconhecimento

O Módulo de Reconhecimento é responsável pela identificação dos usuários no ambiente. Para isto é utilizada a imagem do usuário (obtida pelo módulo de rastreamento) de onde se obtém a face do mesmo. Desta forma o reconhecimento pode ser realizado sem atuação explícita ou intrusão nas atividades do usuário. Seu funcionamento consiste nos seguintes passos:

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 [Ma 2007, P. Viola 2001] cuja a implementação é encontrada na biblioteca OpenCV [Bradski 2000] (Open Source Computer Vision). 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. O reconhecimento facial foi desenvolvido utilizando a técnica *Eigenfaces* [Hewitt 2007, Korting and Filho]. 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.

O reconhecimento facial inicia-se projetando a imagem no subespaço através do método PCA [Belhumeur et al. 1996] 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. Para o cálculo da distância é utilizado tanto a distância Euclidiana como o Mahalanobis [Perlibakas 2004]. Conforme exposto na Sessão 5 foi observado que a utilização de ambas apresentou resultados mais satisfatórios que ambas isoladamente.

4.3. Módulo de Registro

Este módulo trata do registro de novos usuários abrangendo desde a obtenção de suas imagens até o treinamento inicial para seu reconhecimento posterior. O cadastro de um novo usuário é composto pelas seguintes etapas:

1. Obtenção das imagens do novo usuário.
2. Processamento das imagens. Consiste em converter as imagens em escala de cinza, recortar a região da face, redimensionar e equalizar para os padrões de tamanho, brilho e contraste desejados.
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.

4.4. Módulo de Integração

O Módulo de Integração é responsável por disponibilizar ao ambiente os dados dos usuários. Para isto o Sistema TRUE disponibiliza um *Driver* de Usuário (*UserDriver*) através do middleware *uOS*. Tal *driver* fornece os seguintes serviços de consulta e eventos:

- Serviços:
 - Consultas as informações dos usuários no ambiente: através dessas consultas, as aplicações tem acesso aos nomes, e-mails, 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 e-mail 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 e-mail do usuário.
- 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.

5. Testes

Com o intuito de verificar o desempenho do Sistema TRUE foram realizados uma série de testes abrangendo os dados de Rastreamento, Localização e Reconhecimento. Tais testes ocorreram no Laboratório LAICO (Laboratório de Sistemas Integrados e CONcorrentes) do Departamento de Ciência da Computação da Universidade de Brasília.

5.1. Rastreamento

Foram realizados teste com o objetivo de verificar a eficiência da detecção e o dimensão da oclusão dentro do sistema. A oclusão era um problema esperado uma vez que o Sistema TRUE utiliza somente um sensor *Kinect* como dispositivo de entrada. Os testes para a detecção 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.

Foram realizados 2 testes para a oclusão. No primeiro 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. No segundo teste foi simulada 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.

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.

5.2. 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. Os testes foram realizados para as coordenada do eixo x e z.

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). 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. Desta forma o sistema apresenta um erro máximo de 3% sobre os valores obtidos. Um valor que pode ser considerado desprezível para a maior parte das aplicações.

Para testar a precisão das coordenadas do eixo x 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 , $\pm 330mm$, $\pm 660mm$, $\pm 990mm$). 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. Chegamos então a um erro máximo de 10% sobre os valores obtidos. Este é um valor mais significativo que o teste anterior mas, ainda assim, pode ser relevado na maior parte das aplicações.

5.3. Identificação

Para testar os resultados de identificação foram realizadas 2 iterações de testes. 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. Para analisar os resultados foram utilizados três percentuais: verdadeiro positivo, quando o sistema identifica o usuário de maneira correta; verdadeiro negativo, quando o sistema identifica o usuário de maneira errada; e falso negativo, quando o sistema não identifica o usuário cadastrado.

No primeiro cenário 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. Observou-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, **15,83%**, foi bem superior a taxa de Falso Negativo, **8,34%**, 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 do 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.

Para esse segundo cenário houve um aumento na taxa de Verdadeiro Positivo de **73,63%** para **82,5%** e redução significativa na taxa de Verdadeiro Negativo de **15,83%** para **5,46%**, 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. Com uma maior variabilidade nos dados de um mesmo usuário temos um melhor treinamento do sistema. Isso torna os valores de *eigenfaces* calculados mais robustos para uma maior diversidade de poses a serem encontradas durante a execução.

6. Conclusão

As informações de contexto são importantes para auxiliar as aplicações em suas decisões, em especial a localização e a identificação, que fornecem as aplicações um maior entendimento do contexto. Como visto, existem poucos projetos que proveem essas duas informações em conjunto.

Conforme vistos nos testes o Sistema TRUE apresentou uma boa acurácia na identificação de seus usuários. Somado a isso obtivemos também um bom resultado quanto a localização e rastreamento do usuário. Tais dados são fornecidos as aplicações através de uma interface conhecida (o *UserDriver*) que permite também a continuidade de notificações mesmo em casos esporádicos de interrupção e oclusão.

Atualmente estamos aprimorando a confiança e continuidade dos dados providos pelo sistema TRUE buscando agregar os dados obtidos através de múltiplos sensores *Kinect*. Desta forma podemos evitar as descontinuidades ocorridas por oclusões bem como aprimorar a confiança do reconhecimento obtido.

Referências

- Belhumeur, P. N., Hespanha, J. P., and Kriegman, D. J. (1996). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *European Conference on Computer Vision*.
- 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>.
- Buzeto, F. N., Filho, C. B., Castanho, C. D., and Jacobi, R. P. (2011). International Journal of Handheld Computing Research (IJHCR). *International Journal of Handheld Computing Research (IJHCR)*, 2:47–64.
- Gomes, A. R. (2007). Ubiquitos – uma proposta de arquitetura de middleware para a adaptabilidade de serviços em sistemas de computação ubíqua. Master's thesis, Departamento de Ciência da Computação, Universidade de Brasília, <http://monografias.cic.unb.br/dspace/handle/123456789/110>.
- 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>.
- Korting, T. and Filho, N. Utilizando eigenfaces para reconhecimento de imagens. *Fundação Universidade Federal do Rio Grande*.

- 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*.
- OpenNI (2010). *OpenNI User Guide*. OpenNI organization.
- 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*.
- Perlibakas, V. (2004). Distance measures for pca-based face recognition. *Pattern Recogn. Lett.*, 25:711–724.
- 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.
- Stiefelhausen, 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., Stiefelhausen, 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+.