

Visgraf Vision and
Graphics
Laboratory

IMPA

Realidade Aumentada Móvel

Carlos Eduardo de Souza

Vision and Graphics Laboratory - Visgraf - IMPA
Estrada Dona Castorina 110
Rio de Janeiro / Brasil 22460-320
tel +55 21 2529 5000
fax +55 21 2512 4115
Email csouza@impa.br

Relatório Técnico IMPA-001

May 2012

Resumo: Este relatório descreve as atividades realizadas por Carlos Eduardo de Souza no Laboratório Visgraf-IMPA, durante o período de Outubro de 2011 a Maio de 2012 sob a supervisão do Professor Luiz Velho. As atividades deram origem a dois módulos de um sistema de realidade aumentada móvel desenvolvido para celulares e assistentes pessoais digitais (tablets) com sistemas operacionais iOS 4 e iOS 5. O primeiro módulo é responsável pela leitura dos sensores de localização e orientação dos dispositivos móveis e o segundo responsável pela inserção de imagens, vídeos, sons e textos em cenas do mundo real. O trabalho foi apoiado pelo CNPq mediante uma bolsa de DTI-1 processo 384905/2011-0.

1 Introdução

Realidade aumentada (RA) pode ser definida como cenas do mundo real acrescidas de objetos virtuais. Em geral, tanto a captura da cena quanto a inserção dos objetos virtuais (augmentations), acontecem em tempo real e tais augmentations se resumem a gráficos, sons, vídeos e dados textuais localizados através de um sistema de localização global-GPS.

O rápido avanço tecnológico dos dispositivos móveis como celulares (smartphones) e assistentes pessoais (tablets), em relação a sua capacidade de processamento, armazenamento de dados, conexão a internet e também em relação aos sensores inseridos nos mesmos, vem promovendo um grande crescimento no número de sistemas baseados em RA desenvolvidos para tais dispositivos. Esses sistemas são denominados sistemas de realidade aumentada móvel (SRAM).

2 Posição e Orientação do Dispositivo

A determinação da posição e orientação do dispositivo é essencial para uma correta inserção, numa cena do mundo real, de augmentations com posições geográficas pré-determinadas (geolocalizados). Atualmente, os receptores de GPS e a combinação de acelerômetro e giroscópio em dispositivos móveis, juntamente com suas respectivas interfaces para programação de aplicativos (API) [1], habilita o desenvolvimento de aplicações que fazem uso das informações de posição, orientação e de movimento do dispositivo, também conhecidas como informações de movimento em seis eixos (up-down, left-right, forward-backward, roll, pitch, yaw) ilustrado na figura abaixo.

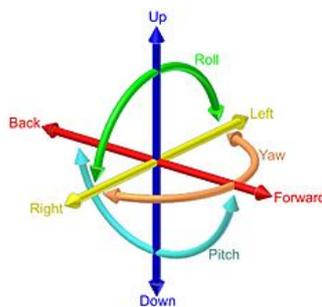


Figura 1: Movimento em seis eixos.

2.1 Posição e Orientação dos Augmentations em Relação ao Dispositivo

Uma vez capturada a posição e orientação do dispositivo, precisamos determinar a posição e orientação relativa ao dispositivo de cada augmentation candidato a ser inserido na cena. A posição do dispositivo é fornecida em geo-coordenadas (latitude, longitude e altitude), a partir dessas, e das geo-coordenadas pré-determinadas do augmentation, calculamos a distância e o bearing entre o dispositivo e o augmentation em questão.

A distância, r , entre o dispositivo e o augmentation é obtida através de uma função intrínseca da API do sistema operacional do dispositivo, em nosso caso particular, iOS, pela função `distanceFromLocation`, onde é levado em conta além das respectivas geo-coordenadas, a curvatura da terra. Para o bearing, θ , aplicamos o seguinte cômputo:

$$\theta = \arctang \left(\frac{\sin(\Delta\text{long})\cos(\text{lat}_a)}{\cos(\text{lat}_d)\sin(\text{lat}_a) - \sin(\text{lat}_d)\cos(\text{lat}_a)\cos(\Delta\text{long})} \right),$$

onde lat_a é a latitude do augmentation, lat_d é a latitude do dispositivo e Δlong é a diferença entre a longitude do augmentation e a longitude do dispositivo. Através da distância e do bearing de um augmentation podemos definir sua posição (em coordenadas cartesianas com centro no dispositivo) da seguinte maneira:

$$\begin{aligned}x &= r\sin(\theta); \\y &= r\cos(\theta); \\z &= \Delta\text{altitude};\end{aligned}$$

onde $\Delta\text{altitude}$ é a diferença entre a altitude do augmentation e a altitude do dispositivo.

As APIs atuais para os sensores de posição e orientação fornecem a orientação do dispositivo através dos ângulos de pitch, roll e yaw, matrizes de rotação e também através de quatérnios, uma boa descrição de quatérnios pode ser encontrada em [3]. Em nosso caso particular, esses dados são obtidos através da classe `CMAAttitude`. A partir desses dados, inicialmente alinhamos os eixos do dispositivo com eixos do sistema onde o eixo y tem a direção/sentido sul-norte (magnético), o eixo x leste-oeste, e o eixo z direção radial do globo terrestre. E em seguida, calculamos a projeção dos augmentations visíveis na área de visão do dispositivo.

3 Inserção de Augmentations Compostos por Gráficos, Vídeos, Sons e/ou Textos em Cenas do Mundo Real

Nossa implementação do processo de inserção de augmentations compostos por gráficos, vídeos, sons e/ou textos em cenas do mundo Real seguiu o padrão model-view-controller (MVC), como é comum nesse processo, e em nosso caso particular, também fizemos uso da estratégia de delegação, ver documentação iOS, conforme ilustrado nas figuras abaixo:

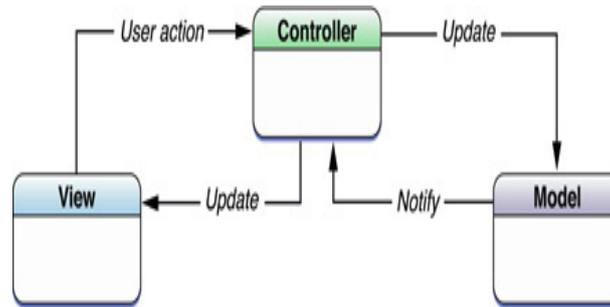


Figura 2: Padrão de Desenvolvimento Model-View-Controller.

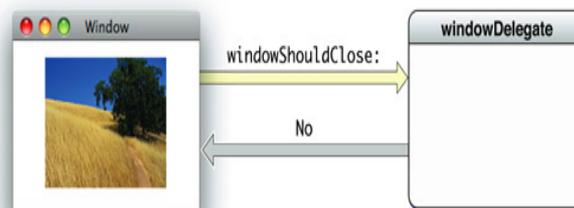


Figura 3: Estratégia de Delegação.

Em particular, em nosso sistema definimos:

- Uma classe denominada *AugmentationManager*, representando o controller do padrão MVC, responsável por responder as mudanças de posição e orientação do dispositivo aplicando as transformações necessárias a cada augmentation.
- Uma classe denominada *AugmentationView*, representando o view do padrão MVC, responsável pela exibição (display) dos augmentations, tratando do carregamento e do posicionamento desses augmentations na tela desses augmentations.
- Uma classe denominada *Augmentation*, representando o model do padrão MVC, responsável por armazenar e disponibilizar os dados referentes aos augmentations, como por exemplo, suas coordenadas geográficas, coordenadas cartesianas e formas de mídias (texto, som, gráficos e/ou vídeo).
- Uma estrutura de delegate-delegating, entre a classe *AugmentationManager* e a classe responsável pela percepção (listener) das mudanças de posição e orientação do dispositivo.

4 Conclusão e Trabalhos Futuros

O objetivo de se familiarizar com o desenvolvimento para dispositivos móveis e de gerar módulos que servissem como base para implementação e testes de novas funcionalidades e

algoritmos foram totalmente atingidos, além disso, os módulos implementados serviram como sub-módulos para o sistema Paprika, em sua versão iOS, um sistema de realidade aumentada para smartphones e tablets. Após uma extensa revisão bibliográfica, temos as seguintes possibilidades como trabalhos futuros:

- Inclusão de animações utilizando OpenGL.
- Utilização de panoramas [5] [4].
- Importação de iluminação do mundo real [2].

Referências

- [1] ALLAN, A. *iOS Sensor Programming: Augmented Reality and Location Enabled iPhone and iPad Apps*. O'Reilly Series. O'Reilly Media, Incorporated, 2012.
- [2] DE CASTRO, T. K. *Sombras Realistas em Realidade Aumentada Móvel*. PhD thesis, IMPA, 2012.
- [3] KUIPERS, J. *Quaternions and Rotation Sequences*. Princeton University Press, 1998.
- [4] MATOS, A., GOMES, J., PARENTE, A., SIERT, H., AND VELHO, L. Um novo ambiente de multimídia e realidade virtual. IMPA, 1997.
- [5] WARRINGTON, C. R. *Markless Augmented Reality for Panoramic Sequences*. PhD thesis, University of Ottawa, 2007.