

ARMsg: Mensageiro Digital usando Realidade Aumentada

Jan van der Haas Habib, Alberto Raposo

Tecgraf, Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro
janvdh69@gmail.com, abraposo@tecgraf.puc-rio.br

Abstract

This paper introduces the ARMsg, an instant messenger using augmented reality. The ARMsg allows that the user send instant messages accompanied by a 3D avatar to be visualized in the camera of the remote user (receiver).

Resumo

Este artigo apresenta o ARMsg, um mensageiro instantâneo utilizando realidade aumentada. O ARMsg permite que o usuário envie mensagens instantâneas acompanhadas de um avatar (modelo 3D) de sua escolha, para serem visualizadas na imagem da câmera do usuário com quem está se comunicando.

1. Introdução

Ambientes Colaborativos de Realidade Virtual (RV) e Realidade Aumentada (RA) abrem uma nova perspectiva para a colaboração em grupo, ao possibilitar que seus participantes interajam através da simulação de um mundo real ou imaginário ou através da manipulação de objetos virtuais no mundo real.

A operação em tempo real característica da RA, reflete-se no uso freqüente de comunicação síncrona nestes ambientes. Nos casos em que os usuários estão co-localizados, a comunicação é realizada diretamente, sem necessidade do ambiente para intermediá-la. No caso de se ter interlocutores fisicamente distantes, áudio-conferência e vídeo-conferência têm sido a solução adotada. Observa-se, porém, que na áudio e na videoconferência, a troca de mensagens não faz uso de objetos virtuais, isto é, nestes casos a Comunicação não é baseada na tecnologia de RA.

Um exemplo clássico sistema de comunicação de RA propriamente dito é apresentado no filme Guerra nas Estrelas de George Lucas. Neste filme, os personagens conversam com interlocutores remotos apresentados como uma imagem virtual de corpo inteiro e em tamanho real. Um sistema de videoconferência RA que

segue esta abordagem é apresentado em [3]. Neste sistema, o rosto do interlocutor remoto é apresentado como um objeto virtual sobre um marcador. Vários interlocutores, móveis ou em desktops, podem participar da conferência simultaneamente. Os marcadores funcionam como monitores virtuais, com a vantagem de poderem ser rearrumados livremente na mesa de trabalho e de serem carregados por usuários móveis.

É notável a evolução dos meios de comunicação nas últimas décadas, no entanto ainda não contamos com um método que possibilite o envio de objetos 3D enquanto nos comunicamos. O presente trabalho visa possibilitar que usuários troquem mensagens e enviem avatares que são visualizados, através da RA, pelo interlocutor em cima de um marcador, através da webcam.

Os ambientes colaborativos de RA estão fortemente associados à Cooperação (no sentido do modelo 3C [5]). Na essência da tecnologia RA está a apresentação e manipulação de objetos virtuais no mundo real, provendo o suporte necessário para que a operação conjunta no espaço compartilhado (Cooperação) ocorra. Mais que isto, este espaço compartilhado é o mundo real e a manipulação dos objetos não é necessariamente delimitada por uma tela de monitor nem precisa ficar presa em torno de uma mesa, abrindo novas possibilidades de suporte à Cooperação.

Do ponto de vista da comunicação, já existem algumas soluções de RA, especialmente para situações síncronas. Para usuários co-localizados, um exemplo é o trabalho de Nakashima et al. [6], que apresenta uma aplicação baseada em interação table-top. Para usuários remotamente localizados, há exemplos de sistemas de videoconferências usando recursos de Realidade Mista, como os trabalhos de Barakonyi et al. [2] e Regenbrecht et al. [9]. O primeiro usa marcadores para inserir objetos virtuais em streams de vídeo convencionais de videoconferência (Figura 1). O segundo é, na verdade,



Figura 1. Screenshot do sistema de videoconferência com RA apresentado em [2].

uma aplicação de virtualidade aumentada, inserindo os streams de vídeo em um ambiente virtual.

O trabalho aqui apresentado usa conceitos de sistemas colaborativos, RA e Computação Gráfica para o desenvolvimento do ARMsg, uma espécie de Windows Live Messenger em RA. A idéia é que os usuários conectados possam trocar mensagens cujo conteúdo sejam objetos (gráficos e textos) virtuais, que serão vistos pelo destinatários sobre o marcador.

Com o programa desenvolvido, espera-se criar um sistema de comunicação tanto síncrono quanto assíncrono (a mensagem é armazenada em arquivo, para posterior visualização) usando recursos de RA, que poderá agregar características multimídia à mensagem (animação, emoção, áudio, etc.).

2. ARMsg

O objetivo principal do projeto é desenvolver uma espécie de “Mensageiro Instantâneo em RA”, onde o remetente cria um avatar e escreve uma mensagem. O ARMsg possui duas funcionalidades diferentes: remetente e destinatário. O remetente é uma área de edição, onde o usuário escolhe o avatar (dentro de uma lista de objetos previamente desenhados no 3DS Max e exportados para .IVE) e redige a mensagem, gerando o arquivo com a animação e texto a ser transmitido. O destinatário é um visualizador de RA que reproduz a

animação (mensagem) gerada sobre o marcador.

Na versão atual, o programa conta com uma biblioteca de modelos 3D que podem ser enviados para o interlocutor. Futuramente, espera-se disponibilizar um repositório de modelos na Internet, para onde os usuários poderão enviar artes 3D próprias, que serão baixadas por todos os usuários ao atualizarem a biblioteca do programa.

O que diferencia este programa das abordagens comuns em comunicação utilizando RA para usuários remotamente localizados, como as apresentadas na seção anterior, é o fato de que no ARMsg, o conteúdo da mensagem é o objeto virtual transmitido entre os usuários. Nas abordagens baseadas em videoconferência, o que é transmitido é o stream de vídeo, sendo os objetos virtuais incluídos localmente. Ou seja, nesses casos, a mensagem em si é o vídeo.

2.1. Arquitetura

A biblioteca OSG [7] é utilizada para construir o grafo de cena e para desenhá-lo, utilizando o OpenGL. A biblioteca ARToolkit [1] é utilizada para capturar a imagem da câmera e fazer o registro espacial, isto é, fazer o reconhecimento de marcadores na imagem da webcam, que serão usados como referência para desenho do objeto virtual. Para que pudesse integrar o grafo de cena do OSG com as funções do ARToolkit, foi necessária a utilização da biblioteca OSGART (OSG

+ ARToolkit) [8]. Um diagrama de como estas bibliotecas se comunicam é mostrado na Figura 2.

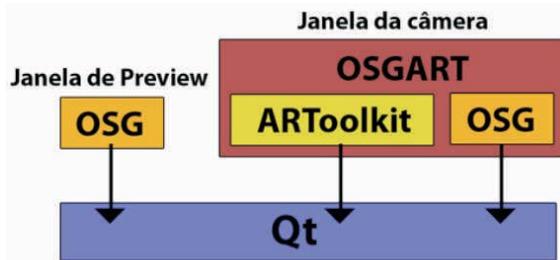


Figura 2. Diagrama de integração entre as bibliotecas.

Resumidamente, o ARToolkit calcula matrizes de transformação para o marcador reconhecido. O OSG usa nós de transformação para posicionar objetos 3D na cena. O OSGART mapeia as transformações do ARToolkit para os nós de transformação do OSG.

A biblioteca Qt é utilizada para a criação da interface gráfica (GUI).

2.2. Características do sistema

O sistema desenvolvido permite que o usuário envie mensagens instantâneas acompanhadas de um avatar (modelo 3D) de sua escolha, para serem visualizadas na imagem da câmera do usuário com quem está se comunicando.

A janela principal do programa (Figura 3) permite que o usuário escolha um avatar, escreva sua mensagem e altere a fonte, cor e tamanho do texto. O usuário pode visualizar o estado em que sua mensagem está após qualquer modificação através da janela de preview. Também na janela de preview, o usuário pode manipular o avatar, para melhor visualização.

Junto com a janela principal, outra janela se abre quando o usuário executa o programa. Esta é a janela da câmera, onde o usuário visualiza as mensagens por ele recebidas em cima de marcadores. Nela, o usuário pode visualizar as mensagens enviadas pelo usuário a quem está conectado ou apenas visualizar algum arquivo de mensagem do ARMsg (.arm) já salvo em seu PC (Figura 4).

2.3. Módulos desenvolvidos

Durante a implementação do sistema, diversos módulos foram desenvolvidos. Esta seção tem por objetivo explicar o funcionamento dos principais módulos desenvolvidos. Todos os módulos herdam da classe base do Qt, a QObject. Isto é necessário para que eles possuam a habilidade de enviar sinais indicando que determinadas ações foram realizadas.

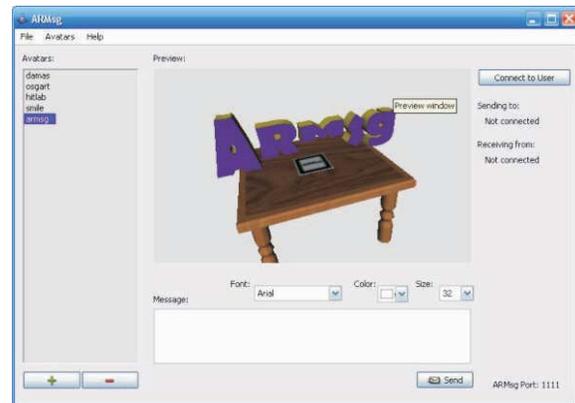


Figura 3. Janela principal do ARMsg.



Figura 4. Janela da câmera do ARMsg.

Estes sinais são associados a slots (funções que são chamadas quando o sinal associado a ela é ativado), para que este faça o procedimento necessário.

O módulo AddAvatarDialog herda da classe QDialog do Qt para a criação de uma janela de diálogo onde o usuário digita o nome do modelo que quer adicionar à sua lista de modelos, sem a necessidade da extensão “.ive”. Só são aceitos modelos que estiverem contidos na pasta padrão de modelos, ou seja, o usuário deve copiar o modelo que quer carregar para esta pasta e assim pode adicionar este modelo para a lista.

O módulo ConnectTo_Dialog herda da classe QDialog do Qt para a criação de uma caixa de diálogo para que o usuário digite o nome ou endereço IP da máquina com a qual quer se conectar. Caso deixe o espaço em branco e clique em OK, o usuário se conecta a ele mesmo, podendo ver as mensagens enviadas por ele na sua janela de câmera.

O módulo OSGWidget herda da classe QGLWidget do Qt para adaptar os eventos do OSG com os eventos de interface do Qt, permitindo assim usar uma janela do

OSG dentro da interface do Qt, como é o caso da janela de preview. Foi implementado usando o sceneview do OSG. Este módulo não foi escrito pelos autores, mas ele é utilizado no módulo descrito abaixo.

O módulo PreviewOSGWidget é responsável pela janela de preview e herda da classe OSGWidget. Possui todas as informações pertinentes à janela de preview. Contém variáveis com seus nós de texto e modelo, bem como funções para criar o grafo de cena e atualizar as informações do texto (fonte, cor, tamanho) e do modelo. Ela reimplementa algumas funções da classe OSGWidget para que se adapte ao propósito da janela de preview, como no caso da função `paintGL()`, que foi alterada para que a cena comece em cima do modelo e não no meio, como é o da OSGWidget.

O módulo AROSGWidget é responsável pela janela da câmera e herda da classe QGLWidget do Qt para adaptar os eventos do OSG com os eventos de interface do Qt. Possui todas as informações pertinentes à janela da câmera. Contém variáveis com seus nós de texto e modelo, bem como funções para criar o grafo de cena e atualizar as informações do texto (fonte, cor, tamanho) e do modelo. Este módulo é baseado no módulo OSGWidget, mas é implementado usando o `osgProducer::Viewer`. O viewer possui um loop próprio de execução, portanto este foi passado para a função `paintGL()` para que o loop `updateGL()` fique redesenhando a tela constantemente e, no caso, atualizando a posição do observador. O grafo de cena da janela da câmera possui nós com o vídeo e o marcador, bem como o texto e os modelos.

O módulo ARMsg herda da classe QMainWindow do Qt para a criação da janela principal do programa. Este módulo possui as funções de manipulação da interface, a parte de conexão cliente-servidor, a parte de persistência em arquivos, bem como todos os caminhos e inicializações padrões.

Para salvar ou enviar mensagens, o aplicativo salva as informações da mensagem em um arquivo com extensão `.arm` (Figura 5).

Assim que o aplicativo executa, ele funciona como um servidor conectado à porta padrão do ARMsg (1111). O cliente é usado para escutar o servidor do outro usuário. Ou seja, cada aplicativo possui um cliente e um servidor. Para o usuário 1 receber mensagens do usuário 2, ele deve se conectar ao servidor do usuário 2 e o mesmo procedimento se aplica para o usuário 2.

O programa foi testado, basicamente, utilizando quatro cenários de teste: usuário salva e abre uma mensagem

```
MSG:
Cow parade!!!!
--end--

FONT:
fonts/Comic Sans MS.ttf

SIZE:
100

COLOR:
1
0.5
1
1

MODEL:
models/vaquinha.ive
```

Figura 5. Formato do arquivo `.arm`

offline, usuário troca mensagens consigo mesmo, usuário troca mensagens através da internet, usuário troca mensagens através de uma rede local. Em todos os casos, o aplicativo funcionou conforme esperado.

3. Conclusões e trabalhos futuros

O projeto aqui apresentado revela uma possibilidade ainda pouco explorada de utilização da tecnologia RA, abrindo um novo campo de estudo que pode ser explorado por diversas vertentes em trabalhos, incorporando RA nas comunicações via internet e até mesmo em outros campos de entretenimento.

Como trabalho futuro, pretende-se modificar a forma de comunicação entre os usuários, abandonando-se o modelo atual de ligação de máquina com máquina para se adotar um servidor central, que receberá todas as mensagens e as encaminhará ao destinatário final, assim como o popular Windows Live Messenger.

Ainda quanto aos avanços que se pretende na arte, espera-se utilizar a biblioteca Cal3D [4], tornando possível que os avatares sejam animados por esqueleto e que possuam movimentos faciais desenvolvidos com Morph Targets, simulando, por exemplo, a fala.

4. Referências

- [1] ARToolkit - <http://www.hitl.washington.edu/artoolkit/>. Acesso em 26/11/2007.
- [2] Barakonyi, I., Frieb, W. and Schmalstieg, D. (2003) Augmented Reality Videoconferencing for Collaborative Work. *Proc. of the 2nd Hungarian Conference on Computer Graphics and Geometry*, Budapest, Hungary, May 2003.

- [3] Billinghamurst, M. and Kato, H. (2002) "Collaborative Augmented Reality". *Communications of the ACM*, Vol. 45, No. 7, July, pp.64 – 70
- [4] Cal3D - <http://home.gna.org/cal3d/> Acesso em 26/11/2007.
- [5] Filippo, D., Raposo, A. B., Endler, M., Fuks, H. (2007) "Ambientes Colaborativos de Realidade Virtual e Aumentada". In: C. Kirner e R. Siscoutto (eds.), *Realidade Virtual e Aumentada: Conceitos, Projeto e Aplicações – Livro do Pré-Simpósio IX SVR*, Cap. 9, pp.168-191. Ed. SBC, Porto Alegre, Brasil.
- [6] Nakashima, et al. (2007). "A 2D-3D integrated tabletop environment for multi-user collaboration". *Computer Animation and Virtual Worlds, Volume 18, Number 1*, February, John Wiley & Sons., pp. 39-56.
- [7] OSG - www.openscenegraph.org. Acesso em 26/11/2007.
- [8] OSGART - <http://www.artoolworks.com/community/osgart/>. Acesso em 26/11/2007.
- [9] Regenbrecht, H. et al. (2003) An Augmented Virtuality Approach to 3D Videoconferencing. *Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)*, pp. 290-291.