

Projeto e Desenvolvimento de um Aplicativo de Geometria Interativa para Dispositivos Móveis

Laís Z. Pedro¹, Simone S. Borges¹, Aparecida M. Z. Lopes¹, João P. T. Souza¹,
Anarosa A. F. Brandão³, Leônidas O. Brandão², Seiji Isotani¹

¹Instituto de Ciências Matemáticas e de Computação

²Instituto de Matemática e Estatística

³Escola Politécnica

Universidade de São Paulo (USP)

{laiszagatti, joaopaulot}@grad.icmc.usp.br,
{sborges, cidazem, sisotani}@icmc.usp.br

Abstract. *Interactive Geometry (IG) considers that the representation and manipulation of geometric spaces, through user-friendly interfaces, can contribute significantly to the teaching-learning processes. Nevertheless, all IG software developed to date is based on the desktop model. In this model it is considered that the interaction between a user and the IG software occurs through the conventional inputs and outputs methods: keyboard, mouse, and a monitor. To build an IG software that can be effectively used in mobile devices such as smartphones and tablets, where inputs are touches and gestures of fingers and the output is a small screen, there is a need to rethink the interface and interaction methods during manipulation of geometric objects. Thus, this paper presents the study, design and development of a GI application for mobile devices based on the Android technology. In this work, we have identified a suitable interface and interaction model to obtain a learning environment aligned with the principles of GI.*

Resumo. *A Geometria Interativa (GI) considera que a representação e a manipulação do espaço geométrico, por meio de interfaces amigáveis, podem contribuir, de modo significativo, para o processo de ensino e aprendizagem da Geometria. Contudo, todos os softwares de geometria interativa desenvolvidos até o momento estão baseados no modelo desktop (computador pessoal). Neste modelo considera-se que a interação entre o usuário e software ocorre através dos padrões de entradas e saídas convencionais: teclado, o mouse e o monitor. Para que a GI possa ser efetivamente utilizada em dispositivos móveis como smartphones e tablets, onde os meios de entrada de dados são toques e gestos de dedos e a saída uma tela gráfica de tamanho reduzido, existe a necessidade de se repensar a interface e os métodos de interação durante a manipulação de objetos geométricos. Assim, este artigo apresenta o estudo, planejamento e desenvolvimento de uma ferramenta de GI para ser utilizada em dispositivos móveis baseados na tecnologia Android. Neste trabalho, identificou-se um modelo adequado de interface e interação para a ferramenta, de modo a se obter um ambiente de aprendizagem alinhado aos princípios da GI.*

1. Introdução

A área da educação tem passado por grandes mudanças nos últimos anos, impulsionada pelos recentes avanços tecnológicos. Recursos computacionais inovadores, novas e eficientes tecnologias de comunicação e as crescentes pesquisas na área educacional contribuem para a reestruturação do modo clássico e tradicional de ensinar e aprender. Outro fator que contribui para esta renovação é a aptidão das novas gerações com relação às novas tecnologias. Tudo isso corrobora com os esforços dedicados ao desenvolvimento e elaboração de novas e mais produtivas ferramentas de ensino nas escolas [Mattar 2009][Klopfer 2012]. Estas mudanças motivam pesquisas nesta área de modo a tornar o processo de ensino-aprendizagem cada vez mais produtivo, dentro e fora da sala de aula.

As tecnologias e dispositivos móveis, tais como celulares, *smartphones* e *tablets* evoluem de maneira cada vez mais rápida quando comparados, por exemplo, aos computadores pessoais. A cada ano são encontradas novas formas de utilizar esses aparelhos, que vão além das finalidades de comunicação originais para os quais foram desenvolvidos. A ampla disseminação destes dispositivos, proporcionados pelos baixos custos e avançados recursos envolvidos, justificam a investigação e pesquisas sobre o potencial uso dos mesmos em ambientes educacionais, principalmente quando se trata de nações em desenvolvimento [Pachler et al. 2009] [Dikkers et al. 2011][Traxler e Kukulska-Hulme 2011].

No Brasil, o Governo Federal iniciou, em 2007, uma campanha de investimento em tecnologias voltadas para a educação. Em uma primeira etapa, o projeto Um Computador por Aluno (UCA) do Governo Federal entregou, às escolas da rede pública, 150.000 *netbooks* [UCA 2012]. Outro programa do governo, previsto para iniciar em 2012, pretende entregar 600 mil *tablets* a professores e alunos do ensino médio das escolas públicas, em todo o Brasil [MEC 2012]. Esse incentivo à disseminação de tecnologias é muito importante para a educação, pois pode contribuir, entre outros fatores, para ampliar o acesso ao conhecimento e viabilizar o desenvolvimento e aplicação de ferramentas educacionais que podem ser utilizadas em dispositivos móveis.

Nas últimas décadas, vários autores se dedicaram a conceituar o processo de aprendizagem, dentre eles o pioneiro Jean Piaget (1999). Para Piaget, a aprendizagem é considerada uma resposta particular, que pode ser complementada por meio das experiências. Piaget postula que, ao nascer, o indivíduo recebe como herança uma série de estruturas biológicas - sensoriais e neurológicas - que permanecem constantes ao longo da sua vida. São essas estruturas biológicas que irão predispor ao surgimento de certas estruturas mentais. Em vista disso, na linha piagetiana, considera-se que o indivíduo carregue consigo duas marcas inatas, que são a tendência natural à organização e à adaptação [Oliveira 2002].

O processo de aprendizagem envolve a assimilação e a acomodação. Na medida em que se participa ativamente dos acontecimentos, se assimila, mentalmente, as informações sobre o ambiente físico e social e se transforma o conhecimento adquirido em formas de agir sobre o meio. O conhecimento assimilado constitui as bagagens de experiências que permite enfrentar novas situações, assimilar outras experiências e formular novas ideias e conceitos. As novas aprendizagens baseiam-se nas anteriores e, assim, a inteligência humana se desenvolve. Aprendizagens simples servem de base a outras aprendizagens mais complexas [Oliveira 2002].

Para desenvolver um software educacional que ofereça apoio para uma aprendizagem ativa, como destacado nos parágrafos anteriores, este trabalho apresenta o processo de criação de um aplicativo para o ensino interativo e colaborativo de geometria, utilizando dispositivos móveis.

Este artigo encontra-se organizado da seguinte forma: na Seção 2, apresenta-se a fundamentação teórica sobre o ensino interativo de Geometria; na Seção 3, encontram-se as considerações sobre desenvolvimento da interface, da estrutura de dados e dos recursos de colaboração; e por fim, na seção 4, encontram-se as considerações finais.

2. Geometria Interativa

A Geometria Interativa (GI), também conhecida como *Geometria Dinâmica*, foi inicialmente usada por Nick Jackiw e Steve Rasmussen. O termo “interativo” do nome pode ser entendido como oposição à forma “estática” com que as construções da geometria tradicional são apresentadas aos alunos [Isotani & Brandão, 2004]. Este tipo de software permite que se crie um ambiente dinâmico e interativo, onde os alunos constroem seu aprendizado, manipulando e alterando objetos geométricos na tela do computador [Brandão et al. 2006].

Segundo Piaget, a mente constrói suas estruturas de conhecimento tomando os dados externos e interpretando-os, transformando-os e reorganizando-os. Pode-se concluir que para aprender é necessário fazer. Infelizmente, existe uma falta de atenção na área de Geometria, o que gera a necessidade de utilizar novas formas e métodos para o ensino da mesma. Segundo Gravina (1996) e Usiskin (1987), o ensino da Geometria recebe pouca atenção tanto no ensino fundamental e médio, quanto no ensino superior. Além disso, atualmente, a Geometria é ensinada de forma tradicional, estática e sem a preocupação de ensinar para a compreensão da base do conteúdo. Por este motivo, a Geometria Interativa se destaca como um método adequado para auxiliar o ensino da geometria, com o uso de tecnologias para explorar todos os recursos computacionais disponíveis. A utilização de softwares de Geometria Interativa permite que o aluno vivencie situações-problema e descubra a solução, fazendo.

A principal diferença entre o aplicativo aqui apresentado e demais softwares que implementam a Geometria Interativa é a forma de interação entre o aluno e o software. Nos softwares baseados em computador *desktop*, como por exemplo iGeom [Brandão et al. 2006], a interação acontece por meio de mouse e teclado. Entretanto, em um aplicativo sensível ao toque, a interatividade proporcionada ao se utilizar os dedos para a manipulação dos objetos geométricos pode fazer com que as ações e o uso das ferramentas se tornem muito mais naturais para o aluno. Na próxima seção será apresentado o processo de desenvolvimento do aplicativo.

3. Desenvolvimento de um Aplicativo Interativo e Colaborativo de Ensino de Geometria para Dispositivos Móveis

O aplicativo está sendo desenvolvido na plataforma Android [Meier 2012] e foi projetado para realizar as funcionalidades básicas de um software de Geometria Interativa. Entre as funcionalidades já desenvolvidas estão a criação, manipulação e remoção de pontos, retas e circunferências; a criação de pontos pertencentes a uma reta ou a uma circunferência; a criação de interseções entre retas e circunferências. Entre as funcionalidades em desenvolvimento estão as que darão suporte ao ambiente de

aprendizado colaborativo. Pretende-se que esse ambiente permita que dois ou mais usuários manipulem objetos geométricos em um mesmo espaço de trabalho, porém utilizando dispositivos distintos.

3.1. Desenvolvimento da Interface

A ideia inicial para o desenvolvimento do aplicativo interativo era que todas as ações fossem realizadas por meio de gestos intuitivos. Uma reta seria concretizada se dois dedos, um em cada ponto, pressionasse a tela por três segundos; Já uma circunferência seria desenhada caso um dos dedos ficasse apoiado na tela e o outro fizesse o contorno, como um compasso. Infelizmente, a ideia inicial não obteve sucesso nos testes de tecnologia e usabilidade, pois diversos problemas foram identificados como, por exemplo, dispositivos que não suportam o multi-toque e a impossibilidade de executar algumas ações facilmente. Para que o software seja compatível com o maioria dos aparelhos disponíveis, optou-se por desenvolver a interação por meio de menus e de botões. Assim, nesta etapa do projeto não haverá operações multi-toque.

As interfaces de diversos aplicativos de desenho para dispositivos sensíveis ao toque foram observadas e analisadas. A partir deste conhecimento e utilizando técnicas de Interação Humano-Computador (IHC), chegou-se a uma solução que utiliza somente uma área mínima da tela do dispositivo (Figura 1). A opção de criar um menu com botões poderia atrapalhar o que é primordial em um aplicativo de desenho em um dispositivo móvel: o tamanho da tela. A solução para esse problema foi criar, no canto inferior esquerdo, apenas um botão que aciona o menu de ferramentas. Este botão mantém, no seu ícone principal, o ícone da ferramenta escolhida, para permitir que o usuário identifique qual ferramenta está selecionada no momento. Também criou-se um botão de remoção de objetos da tela. Como a remoção de objetos só pode ocorrer quando um objeto está selecionado, este botão aparece apenas quando um ponto (ou outro objeto) está selecionado. Desta forma, a opção de remoção irá aparecer apenas quando existir necessidade, otimizando o espaço em tela. Na Figura 1, à direita, é possível visualizar o botão de remoção no canto inferior direito. O objeto selecionado é o ponto verde com maior raio na figura. Ao clicar neste botão, o objeto selecionado será removido, assim como todos os outros que dele dependem.

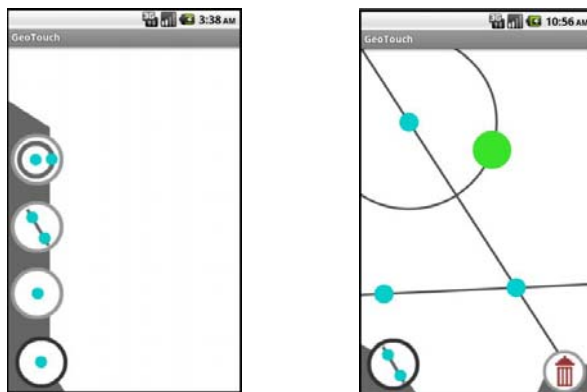


Figura 1. À esquerda, interface com o menu de ferramentas expandido. À direita, ponto selecionado (destacado em verde) e com a opção de remoção no menu inferior direito.

Para desenvolver a estrutura interna de um aplicativo de Geometria Interativa, o primeiro passo foi criar os objetos-base. Dentre os objetos-base utilizados neste trabalho existe um que é a base de todos: o **ponto**, o qual foi estudado, cautelosamente, para a

melhor manipulação possível. Considerando-se que todos os dispositivos móveis que serão utilizados deverão ser *Touch Screen*, ou seja, sensíveis ao toque, o dedo seria o manipulador dos objetos. O desenho do ponto na tela foi elaborado com precisão, em um tamanho adequado, de forma que o usuário não tivesse dificuldades em manusear os objetos. O desenho para representar o ponto na tela é uma figura circular, que o usuário consegue manipular, a partir do toque, e arrastando o dedo pela tela.

A base do aplicativo é formada por duas classes: *GeoTouchActivity* - a qual gerencia todos os eventos de toque e contém todos os objetos do aplicativo - e a classe *Ponto* - que estende a Classe *View* do Android, para que os objetos dessa classe possam ser adicionados como filhos da classe *GeoTouchActivity*, assim exibidos na tela. Com a base do aplicativo pronta, o próximo passo foi adicionar as funcionalidades para a criação de retas e circunferências. Para desenhar uma reta ou uma circunferência são necessários, pelos menos, dois pontos. Com a utilização da funcionalidade *criar pontos*, foram criadas outras duas classes: a classe *Reta* e a classe *Circunferência*. Ambas possuem a mesma estrutura e recebem, em seu construtor, dois pontos para que possam ser desenhadas. A única diferença é a equação e a função de desenho. No caso da reta, existe um problema: nenhuma classe padrão do Android possui uma função que permita desenhar uma reta “infinita”. Para resolver este problema foi necessária a realização de alguns cálculos.

Para uma reta R , criada a partir dos pontos $P1 (x_1, y_1)$ e $P2 (x_2, y_2)$. É possível obter a equação de R : $y = ax + b$, Onde:

$$a = \frac{(y_2 - y_1)}{(x_2 - x_1)} \quad \text{e} \quad b = y_1 - \frac{(y_2 - y_1)}{(x_2 - x_1)} x_1$$

Por meio dessas equações foi possível criar dois métodos para a classe *Reta*, chamados $FX(float y)$ e $FY(float x)$. A FX retorna o valor x de um ponto, dado seu valor y ; e o método FY retorna o valor y de um ponto, dado seu valor x . Esses métodos usam a equação da reta para obter esses valores. Para que seja possível desenhar um segmento de reta maior que a tela é preciso achar as coordenadas de pontos da reta R , as quais não estão visíveis ao usuário. Para isso, as coordenadas de $P1$ e $P2$ são comparadas para descobrir a inclinação da reta e, assim, determinar as coordenadas dos pontos, utilizando-se os métodos FX e FY . A Figura 2 ilustra a lógica utilizada.



Figura 2. Cálculo dos pontos da reta.

Com os valores dos pontos P3 e P4 é possível desenhar um segmento de reta que será grande o suficiente para cruzar a tela e dar a sensação de que a reta criada é infinita (Figura 3 – à esquerda). Para desenhar circunferências, o único cálculo necessário é para gerar o raio, que é a distância entre os pontos P1 e P2 (Figura 3 – à direita).

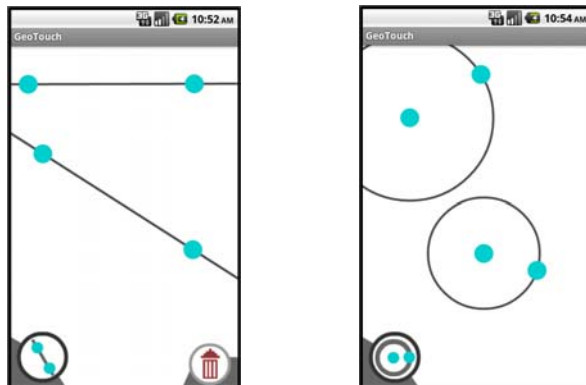


Figura 3 – Exemplo de retas e circunferências no aplicativo.

A criação de retas e circunferências é dependente da criação de pontos. Assim, a remoção de pontos deve ser tratada de uma forma especial. Se uma reta for criada e, depois, um de seus pontos-base for removido, não existe uma forma dessa reta continuar a existir, visto que não se pode desenhar uma reta a partir de apenas um ponto. Dessa forma, foi necessário criar uma estrutura de relacionamento, entre os objetos do aplicativo, para garantir que os objetos dependentes de outros objetos sejam removidos de maneira consistente. Este relacionamento pode ser analisado como um grafo. Na Figura 4, à esquerda, é possível observar uma reta R dependente de pontos A e B . Caso A ou B sejam removidos, a reta R também será removida. A complexidade aumenta quando se criam novas retas ou circunferências, utilizando-se novos pontos ou pontos já existentes. Esta linha de raciocínio orienta para mais uma funcionalidade do aplicativo: pontos pertencentes às retas ou circunferências. Esses pontos, por sua vez, seriam dependentes dos objetos para os quais foram criados.

No exemplo, à direita, da Figura 4, se criou um ponto C **sobre** a reta R (ou seja, C pertence a R). Neste caso, se o ponto A ou B for removido, a reta R será removida e, consequentemente, o ponto C , que é dependente da reta R , também será removido. Caso houvesse mais dependências na hierarquia abaixo de C , estes também seriam removidos.

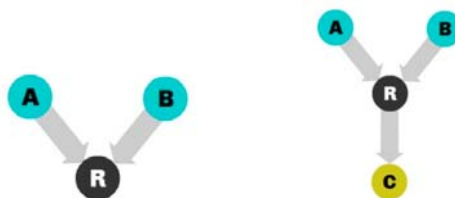


Figura 4. Estrutura de Relacionamento.

Na prática, a criação de um ponto na tela sensível ao toque mostrou-se um procedimento mais complicado que o previsto originalmente. Devido ao tamanho reduzido da tela, a ação de criar com precisão um novo ponto em uma reta já existente ficou comprometida. A solução encontrada foi a criação do novo ponto em uma área não ocupada da tela, e sua posterior movimentação para cima da reta que se deseja

incluir o novo ponto. Dessa forma, ao invés de criar o ponto diretamente dentro da reta, o usuário manipula o ponto para cima da reta desejada e então faz com que ele “pertença” a essa reta (ou seja, fique sobre a reta). Para implementar tal solução foi preciso criar um cálculo de colisão entre o ponto sendo movido e os outros objetos geométricos (retas e circunferências, por exemplo) do espaço de desenho. Para isso foi criado o método *isTouched* nas classes Reta e Circunferência para identificar que dado um ponto $P(x,y)$ existe ou não colisão com a reta ou circunferência. A Figura 5 mostra como a colisão entre ponto e reta é detectada.

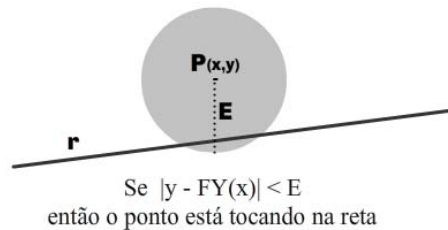


Figura 5. Colisão entre ponto e reta.

A margem de erro E existe para garantir que exista uma proximidade com a reta, pois, a interação, com utilização do toque, não é muito precisa e, portanto, seria difícil conseguir posicionar um ponto exatamente em cima de uma reta. Para a circunferência, foi utilizada a mesma ideia (Figura 6), porém a formula matemática é diferente.

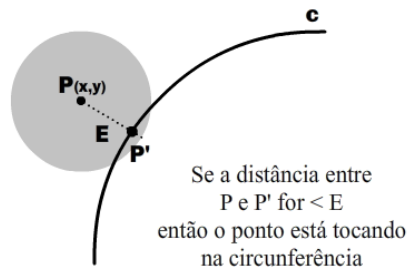


Figura 6. Colisão entre ponto e circunferência.

Com a solução implementada, é possível criar pontos e transformá-los em pontos dependentes de uma reta ou uma circunferência. Para facilitar o uso dessa funcionalidade, foi definida uma cor diferente para o ponto que é dependente de uma reta ou circunferência. Na Figura 7, o ponto posicionado no centro da reta e o ponto na parte superior da circunferência são pontos dependentes.

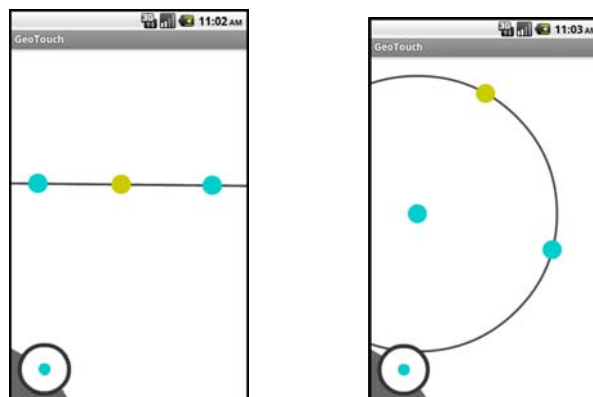


Figura 7. Ponto dependente na reta e na circunferência.

A última funcionalidade implementada no aplicativo foi a possibilidade de fazer interseções entre retas e circunferências. A interseção entre retas e circunferências é representada por um ponto. Para conseguir criar uma interseção entre duas retas é necessário que um ponto pertença a ambas as retas. Portanto, é preciso viabilizar a criação de pontos pertencentes a dois objetos geométricos, por meio da utilização da estrutura de grafos definida neste trabalho, quando uma reta R é criada a partir dos pontos A e B ; se existir uma outra reta S criada a partir dos pontos C e D , a interseção entre as retas R e S é um ponto E que pertence às duas retas, conforme mostra a Figura 8.

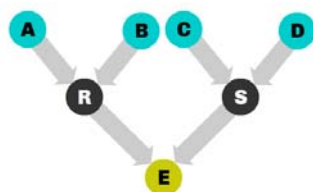


Figura 8. Exemplo de interseção.

3.3. Funcionalidade em Desenvolvimento: Colaboração

O desenvolvimento de software se beneficia de cada avanço no campo do desenvolvimento tecnológico. As pesquisas sobre ensino colaborativo são beneficiadas por esses avanços e ganham força com a rápida evolução da internet [Stahl et al. 2006]. Contudo em um ambiente virtual de aprendizagem, estes avanços não são o único ponto de atenção. O trabalho em conjunto propicia um processo cognitivo que é compartilhado entre os membros de um grupo e não apenas em um indivíduo [Flavell et al. 1999] [Branco 2010]. Neste contexto, a aprendizagem de forma colaborativa é vista como parte de uma estratégia na educação onde dois ou mais indivíduos adquirem seus conhecimentos de forma compartilhada, por meio de discussões, reflexões e atitudes tomadas em conjunto [Comasseto 2006] [Isotani & Mizoguchi, 2008]. Pretende-se com a ferramenta apresentada neste trabalho oferecer um ambiente colaborativo para o aprendizado de geometria, que suporte a comunicação entre alunos e professores por meio dos dispositivos móveis.

A tecnologia utilizada para o desenvolvimento das funcionalidades de colaboração é a mesma interface de comunicação implantada nos dispositivos móveis. Atualmente, há duas formas de realizar esta comunicação - o *bluetooth* e *wireless*. Há que se considerar que este aplicativo será utilizado em escolas, o que sugere a tecnologia *bluetooth* como a mais adequada, tendo em vista a ausência de *wireless* na maioria das escolas brasileiras. A utilização do *bluetooth*, como forma de comunicação entre os dispositivos, propicia a colaboração entre alunos e professor, de forma que os alunos consigam ver em seus dispositivos o que o professor está fazendo e, de forma interativa, colaborar com a atividade, proporcionando que professor e aluno trabalhem juntos na produção do conhecimento.

O maior desafio neste trabalho é unir a colaboração com a interação dinâmica para o auxílio da aprendizagem em Geometria. O espaço virtual de aprendizagem apresenta características distintas dos demais. Dentre estas características, a simulação, a vivência de um problema e a demonstração virtual de uma teoria integram ainda mais a interatividade e colaboração, transformando o conhecimento adquirido em aprendizado.

4. Considerações Finais

Neste trabalho foram apresentados os desafios propostos e soluções encontradas para desenvolver um aplicativo para ensino de Geometria em dispositivos móveis, na plataforma Android. O desenvolvimento de softwares educativos para dispositivos móveis é um campo de pesquisa em expansão, que acompanha a evolução das tecnologias para utilizar cada vez mais recursos e oferecer maior flexibilidade para o processo de ensino-aprendizagem. Desenvolver softwares educativos para dispositivos móveis não é uma tarefa trivial, já que há necessidade de identificar métodos de interação e soluções computacionais com a utilização de tecnologias que se encontram em constante atualização e evolução.

As funcionalidades desenvolvidas até o momento implementam as funções básicas da Geometria Interativa como, por exemplo, a criação e remoção de pontos, retas e circunferências; a manipulação dinâmica dos objetos criados; e a possibilidade de criar interseções simples tais como pontos sobre objetos. Vários desafios de pesquisa e desenvolvimento foram superados. Pode-se destacar, entre eles, a criação de uma interface intuitiva para construção de objetos geométricos, o desenvolvimento de uma estrutura de dados que atenda aos requisitos da geometria interativa, e a resolução de problemas de interação relacionados com a criação de objetos geométricos, por meio da utilização de telas sensíveis ao toque. Como trabalhos futuros, pretende-se implementar mais funcionalidades no aplicativo, de forma a promover aprendizagem colaborativa entre usuários por meio de múltiplos dispositivos. Serão realizados experimentos para validação da ferramenta e dos conceitos aqui apresentados.

Agradecimentos

Os autores gostariam de agradecer o apoio financeiro proporcionado por MCT/CNPq/CAPES/MEC-SEB (processo: 550449/2011-6) e CNPq (processo: 476566/2011-8). Agradecemos também à Ercília M. B. Zagatti Pedro pelas valiosas discussões e sugestões.

Referências

- Branco, S. E. (2010) Possibilidades de Interatividade e Colaboração Online: Uma proposta de formação continuada de professores de matemática. Dissertação de mestrado, UFPR (Mestrado em Educação). Curitiba.
- Brandão, L. O., Isotani, S., Moura, J. G. (2006) Imergindo a Geometria Dinâmica em Sistemas de Educação a Distância: iGeom e SAW. *Revista Brasileira de Informática na Educação*, 14(1), 41-49.
- Comasseto, L. S. (2006) Novos Espaços Virtuais para o Ensino e Aprendizagem a Distância: Estudo da aplicabilidade dos desenhos pedagógicos. Tese de Doutorado, UFSC (Doutorado em Engenharia da Produção). Florianópolis.
- Dijkers, S.; Martin, J.; Coulter, B. (2012) *Mobile Media Learning: Amazing uses of mobile devices for learning*. Etcpress, USA.
- Flavell, J. H.; Miller, P. H.; Miller, S. A. (1999) “Desenvolvimento Cognitivo”, 3ª Edição. Artes, Porto Alegre.

- Gravina, M. Alice. (1996) Geometria Dinâmica: Uma nova abordagem para o aprendizado da geometria. Anais do Simpósio Brasileiro de Informática na Educação.
- Isotani, S.; Brandão, L. O. (2004) Automatizando o processo de geração e correção de exercícios no iGeom. Anais do Simpósio Brasileiro de Informática na Educação, 328-337.
- Isotani, S.; Mizoguchi, R. (2008) Theory-Driven Group Formation through Ontologies. In: International Conference on Intelligent Tutoring System. Lecture notes in Computer Science. Berlin Heidelberg: Springer-Verlag, 5091, 646-655.
- Klopfer, E. (2008) Augmented Learning: Research and design of mobile educational games. MIT Press, USA.
- MEC. Ministério da Educação e Cultura. (2012) Tecnologia na Educação: Ministério distribuirá tablets a professores do ensino médio. Disponível em: http://portal.mec.gov.br/index.php?option=com_content&view=article&id=17479. Acesso em: 31 de jul. de 2012.
- Meier, R. Professional Android Application Development. John Wiley & Sons, 2012.
- Oliveira, M. K. (2002) Pensar a Educação, Contribuições de Vygotsky. In: Piaget-Vygotsky: Novas contribuições para o debate. 6ª ed. Ática. São Paulo.
- Pachler, N.; Bachmair, B.; Cook, J.; Kress, G. (2009) Mobile Learning: Structures, Agency, Practices. Springer, USA.
- Piaget, J. The psychology of intelligence. Taylor & Francis, 1999.
- Mattar, J. (2009) Games em Educação: Como os nativos digitais aprendem. Person, São Paulo.
- Stahl, G., Koschmann, T., & Suthers, D. (2006). Computer-Supported Collaborative Learning. In R. K. Sawyer (Ed) *Cambridge Handbook of the Learning Sciences*. (pp. 409–426). UK: Cambridge University Press.
- Souza, J. P. T. (2011) Projeto e Desenvolvimento de um Aplicativo de Geometria para Android. Monografia de Conclusão de Curso. Instituto de Ciências Matemáticas e de Computação (Bach. Ciência da Computação), Universidade de São Paulo. São Carlos
- Traxler, J.; Kukulska-Hulme, A. (2005) Mobile Learning in Developing Countries: The Knowledge Series. Commonwealth of Knowledge, Canada.
- UCA. Um Computador por Aluno. Disponível em: <http://www.uca.gov.br/institucional/projeto.jsp>. Acesso em: 30 de julho de 2012.
- Usiskin, Z. (1987) Resolving the Continuing Dilemmas in School Geometry. In: Lindquist, M. M.; Shulte, A. P. (eds). *Learning and Teaching Geometry. National Council of Teachers of Mathematics*, USA.