

Plataforma para jogadores com restrições motoras nos membros superiores

Paulo Victor de Oliveira Leal, Rosilane Ribeiro da Mota
Instituto de Ciências Exatas e Informática
PUC Minas
Belo Horizonte, Brasil
pvoaleal@sga.pucminas.br, rosilane@pucminas.br

Abstract—In this paper it was analyzed and developed a software that - through means of voice recognition - allows for users with physical impairments or disabilities to interact with video games. The system developed has the purpose to act in single player 2D platform games in personal computers and to be completely basic user level in it's design, providing the experience of interacting with the most common control devices for video games. The development was divided in two phases: command generation and implementation. The voice commands utilized by the software were based in 2D games with varied game mechanics so the maximum number of actions possible could be obtained, thus making it possible to generate easy to pronounce generic commands that work with said games, for example “DIREITA”, “ESQUERDA”, “PULAR”, among others (portuguese for right, left and jump respectively). A software with a simple graphical user interface and usability resembling a gaming joystick was implemented, with each button associated to a voice command. The system was validated through tests with users that both possessed and didn't possed some form of physical disability. An average of 70% accuracy was obtained in the voice recognition for the commands, managing to meet the need for both mouse and keyboard and joystic controller schemes for those interested in gaming.

Resumo—Neste artigo foi apresentado o desenvolvimento e análise de um *software* que, por meio do uso de reconhecimento de voz, permite usuários com deficiência físico-motora a interação com jogos digitais. O sistema desenvolvido tem o propósito de servir em jogos do tipo plataforma 2D, single-players para computadores, e que seja totalmente ao nível de usuário, provendo a experiência de interação com controles comuns. O desenvolvimento foi dividido em etapas: geração de comandos e implementação. Os comandos de voz foram gerados baseando em jogos 2D, com diferentes mecânicas para obtenção do máximo de ações possíveis, tornando possível gerar comandos genéricos e de fácil pronúncia que sirvam para todos esses jogos, por exemplo: “DIREITA”, “ESQUERDA”, “PULAR”, dentre outros. Foi implementado um software com interface de usuário e usabilidade simples, tendo como base uma joystick, onde em cada botão foi associado um comando. A validação do sistema foi efetuada por meio de testes com pessoas com e sem deficiência. Foi obtido durante os testes a média de 70% de precisão no reconhecimento dos comandos, conseguindo agradar e suprir a necessidade de mouses/teclados e joysticks dentre o público com interesse em jogos.

Keywords-reconhecimento de voz; acessibilidade; deficientes-físicos; jogos digitais;

I. INTRODUÇÃO

Jogos digitais ou eletrônicos são *softwares* geralmente voltados para o entretenimento de todas as idades [1]. A participação destes no mercado, tem crescido nos últimos anos, principalmente depois do lançamento de plataformas que facilitam a compra e instalação, como: Steam¹, Origin², dentre outras [2]. Devido a esse crescimento, os jogos têm se tornado um entretenimento não só de crianças e adolescentes, mas de adultos e até de idosos [3]. Porém, parte das pessoas são impedidas de jogar devido à deficiências físico-motoras. Existindo esse problema, fundações e organizações, como a *The Able Gamers Foundation* [4], foram criadas visando desenvolver tecnologias inclusivas para auxiliar essa parte da sociedade.

Essas tecnologias têm como conteúdo a melhoria de jogabilidade e o desenvolvimento de alternativas para a entrada de dados do controle, ou seja, a substituição de *joysticks* dos *consoles* e do *mouse*/teclado de computadores. Uma maneira de aplicar a substituição é realizar a entrada de dados através do uso de Sistemas de Reconhecimento de Voz Automático (ASR), com o qual é possível remover a completa necessidade de controles, passando a usar somente a fala do usuário.

O IBGE [5] aponta em seu último censo publicado do ano de 2010 que o Brasil tem cerca de 13 milhões de pessoas com deficiência motora. Com base nesse número elevado, o objetivo desse trabalho é desenvolver um sistema de reconhecimento de voz automático, que possa ser usado em jogos de computador do gênero de plataforma 2D, para um único jogador (*single-player*). O sistema deve ser de baixa complexidade para o usuário, devido à limitação dos comandos de voz. Dessa forma, pode-se ampliar a quantidade de jogadores brasileiros e permitir a inclusão de muitos indivíduos.

II. TRABALHOS RELACIONADOS

O artigo do autor Fava [6] demonstra um jogo desenvolvido pelo próprio autor onde é utilizado o microfone como controle, porém, utilizando o sopro para jogar. O autor

¹<http://store.steampowered.com/>

²<https://www.origin.com/>

visou desenvolver um jogo acessível para deficientes motores onde a movimentação é extremamente limitada. Esse jogo não foi desenvolvido com o propósito de ser completo e sim para avaliar o sopro como um controle para jogos, fornecendo não só uma alternativa a entrada de dados, como também oferecer um exercício para o sistema respiratório. Nesse artigo é possível observar como diferentes controles para jogos estão sendo estudados e desenvolvidos para tornar capaz a inclusão das pessoas com deficiência para o universo dos jogos.

Ao utilizar sistemas de reconhecimento de voz, Mustaquim [3] desenvolveu um sistema no nível de *hardware* que funciona como um *joystick* totalmente comandado por voz. O desenvolvimento foi motivado devido à análise da quantidade e às dificuldades enfrentadas por deficientes e idosos para conseguirem jogar. Dentre as pesquisas do autor, nos EUA no ano de 2003, as pessoas com qualquer tipo de deficiência somavam quase 50 milhões de pessoas. Sendo um sistema para todos os gêneros de jogos, o problema enfrentado foi a dificuldade de mapear todos os comandos para os diversos jogos e jogadores. Apesar de utilizar reconhecimento de voz e jogos digitais, o *hardware* desenvolvido nesse estudo, não é acessível para todos os usuários por demandar custos e o reconhecimento de voz somente trabalha com comandos em inglês.

O *framework* desenvolvido pelos autores López, Corno e Russis [7] foi aplicado durante o desenvolvimento do jogo, funcionando como uma camada acoplada ao jogo ou como um complemento para a *hud*. Os mini jogos em que a aplicação foi testada, foram desenvolvidos pelos próprios autores, e foram nomeados como: *One Switch Demo*, *One Switch Ladybugs* e *One Switch Invaders*. O *framework* teve o objetivo de substituir o método de entrada de dados, com a inclusão de um novo elemento gráfico nos jogos. O controle para esse sistema utilizou um único botão (conhecido como *one-switch*) e no jogo foi incluído um relógio onde o ponteiro apontava para onde o clique era efetivado. Os autores conseguiram resultados com taxa de erro satisfatória (abaixo de 50%), os participantes envolvidos tinham deficiências muito severas. Esse *software* porém deve ser aplicado a jogos extremamente simples: com um único clique para efetuar a ação, 2D, e onde as ações devem ser praticamente mover um objeto.

III. METODOLOGIA

O desenvolvimento do trabalho foi realizado nas seguintes etapas: coleta de dados, elaboração de comandos, validação do sistema e testes com usuários.

A. Coleta de dados

Os jogos avaliados no trabalho são de plataforma 2D, como dito anteriormente e são apresentados no Quadro I, separados pelas diferentes plataformas e o componente para entrada de dados. Os jogos selecionados foram escolhidos

por serem relativamente de fácil jogabilidade e compreensão, o que facilitou os testes com usuários.

Quadro I
JOGOS AVALIADOS

Nome	Plataforma	Entrada de dados
Super Mario	Super Nintendo	Controle
Megaman X	Super Nintendo	Controle
Sonic Mania	PC	Controle/Teclado
Shovel Knight	PC	Controle/Teclado
Mark of the Ninja	PC	Controle/Teclado

Fonte: Elaborado pelo autor

A mecânica dos jogos de plataforma geralmente se resume ao comando de movimentação, pulo, ação (ou usar habilidade) e controle (pausar, dentre outros). A coleta desses comandos é importante para manter os comandos de voz fiéis e semelhantes aos tradicionais usados em jogos por meio dos controles e assim não alterar a essência da jogabilidade. Através de testes (aproximadamente uma hora) pelo autor com cada um dos jogos do Quadro I, foi construída o Quadro II, contendo uma enumeração de cada comando possível e a quantidade de jogos em que ele aparece.

Quadro II
MECÂNICAS POR JOGO

Tipo	Comandos							
Movimento	Direita [Todos]	Esquerda [Todos]	Cima [Todos]	Baixo [Todos]	Correr [Super Mario, Mark of the ninja, Shovel Knight]	Agachar [Todos]		
Camera	Direita [Mark of the ninja]	Esquerda [Mark of the ninja]	Cima [Mark of the ninja]	Baixo [Mark of the ninja]				
Ação	Pular [Todos]	Pular girando [Super Mario]	Usar item [Mark of the ninja]	Impulso [Sonic Mania]	Carregar ataque [Megaman X, Super Mario]	Ataque simples [Mark of the ninja, Shovel Knight]	Ataque forte [Mark of the ninja]	Defender [Mark of the ninja]
Itens	Item 01 [Mark of the ninja]	Item 02 [Mark of the ninja]	Item 03 [Mark of the ninja]	Item 04 [Mark of the ninja]				
Controle	Menu [Todos]	Mapa [Mark of the ninja]						

Fonte: Elaborado pelo autor

B. Elaboração de comandos

O reconhecimento de voz é lento para frases longas e em jogos os comandos devem ser realizados de maneira rápida. Portanto, os melhores comandos são os curtos, simples e de fácil pronúncia.

De modo a se tornar mais intuitivo ao jogador, os comandos elaborados para teste foram pensados de maneira a ser executados como ordens ao protagonista do jogo e portanto foram escolhidos comandos de boa dicção. Priorizando isso os comandos de *ação* e alguns de *controle* foram criados usando verbos no infinitivo, facilitando o uso para os usuários.

Foram apresentados no Quadro III os comandos mapeados por tipo. Esses comandos tomaram como base o Quadro II e foram adaptados para serem executados por voz. Percebe-se que os comandos de *controle* aumentaram, já que passaram a controlar a tecla ativa para comandos iguais, por exemplo os de movimento e câmera. Para os do tipo *ação*, alguns comandos foram unidos para criar um mais genérico como dito anteriormente, o comando “IMPULSO” e “PULAR GIRANDO” foram unidos em “HABILIDADE” e os comandos

mais usados preencheram as outras teclas. Os comandos de itens que estão nas *setas* utilizam sua própria direção.

Quadro III
COMANDOS PROPOSTOS NESTE TRABALHO

Tipo	Comandos						
Movimento	Direita	Esquerda	Cima	Baixo			
Câmera	Direita	Esquerda	Cima	Baixo			
Ação	Pular	Habilidade	Atacar	Correr			
Setas	Direita	Esquerda	Cima	Baixo			
Controle	Menu	Selecionar	Seguir	Parar	Analógico	Gatilho	Câmera

Fonte: Elaborado pelo autor

Os comandos e interface de usuário escolhidos foram validados através de pesquisa *online* aberta ao público com conhecimento em jogos e computação, por meio de um formulário do *Google Forms*, no qual 30 pessoas participaram. Dentre o resultado, 80% das respostas apontaram que os comandos escolhidos foram apropriados para cada tecla. Dentre os restantes, ou ofereceram pequenas alterações que foram aplicadas ou não sugeriram nada.

Para facilitar o aprendizado e entendimento do sistema foi utilizado uma imagem característica de um controle de *Xbox*. Para cada botão do controle padrão, foi adicionado uma caixa de texto com o devido comando que o representa. Na seção inferior ficam os comandos de *controle* onde é possível alternar o tipo de comando a ser executado. Por exemplo, executando o comando “ANALOGICO” os botões direcionais, da câmera e os do gatilho ficam desativados e portanto os comandos “CIMA”, “BAIXO”, “DIREITA” e “ESQUERDA” são executados simulando o analógico.

C. Validação do sistema

O *software* foi implementado para plataforma *Windows*, utilizando a linguagem de programação C# utilizando o padrão *Windows Forms*, onde cada tela é um *Form* diferente, junto com a *Microsoft Speech API* para o reconhecimento de voz.

A *joystick* criada mapeia um comando de voz a uma ação do *Windows*, seja ela o pressionar de uma tecla ou o movimentar do *mouse*, e essa ação é realizada sobre o jogo executado no momento. O usuário inicia o sistema e logo após, algum jogo. A partir daí, os comandos já podem ser executados.

O sistema foi desenvolvido com o objetivo de ser viável financeiramente. As alternativas existentes em mercado atualmente são extremamente caras, como o controle de boca que custa entre US\$ 399,00 a US\$ 549,00 ou o controle de iris que custa em torno de US\$ 149,00 e demandam *hardwares* específicos. O sistema proposto, demanda somente um microfone que tem preços muito inferiores (aproximadamente R\$50,00).

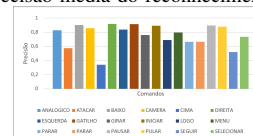
1) *Reconhecimento de voz*: A API do *Google* [8] apresentou o melhor desempenho, porém ela é paga e não pode ser aplicada ao projeto. A API do *Coruja* [9], desenvolvida pela Universidade do Pará, é focada no português brasileiro

e apresentou um desempenho satisfatório, com média de tempo de resposta de 15 ms. Entretanto, a dificuldade de instalação e a quantidade de pré-requisitos tornou inviável o uso para o público alvo. Sendo assim, a escolhida foi a API da *Microsoft* [10]. Essa API oferece um procedimento para reconhecimento parcial do comando e ao aplicá-lo o reconhecimento se tornou quase instantâneo (média de 5 ms), apesar de a taxa de erros para alguns comandos, como pode ser visto na Figura 1, ter aumentado. Além disso, seu uso é livre e a instalação é muito simples. Para realizar o *input* no jogo foi utilizada a biblioteca *Windows Input Simulator* desenvolvida por Noonan. [11]

Ao observar a Figura 1, é possível ver que grande parte dos comandos alcançaram aproximadamente 70% de precisão. Essa depende da qualidade do microfone, no caso foi utilizado um mediano. O gráfico foi gerado a partir da execução de cada comando cem vezes e assim cada coluna representa a porcentagem da quantidade vezes em que o reconhecimento foi correto. Os comandos com baixa precisão, apresentam essa situação devido à API ainda não conseguir alta precisão com comandos sibilantes da língua portuguesa, porém, são pouco usados, o que não interferiu na utilização. Essas palavras foram mantidas por serem mais intuitivas para os usuários e portanto alternativas não foram aplicadas para os testes efetuado, mas é possível alterar todos os comandos do modo como o usuário preferir.

O código do sistema segue no *GitHub*¹. É possível realizar o download dos arquivos fontes como as *builds* para utilizar. Para instalar é necessário somente extrair e abrir o arquivo executável. Ao inicializar o sistema já está pronto para uso e para receber comandos.

Figure 1. Precisão média do reconhecimento por comando



Fonte: Elaborado pelo autor

IV. RESULTADOS

Os testes do sistema foram efetuados por meio de pessoas com deficiência nos membros superiores e pessoas sem deficiência. Os participantes foram contactados através de um canal de recrutamento nas redes sociais, os interessados entraram em contato para participar dos testes. Todos os usuários testaram o sistema por pelo menos 10 min, jogando um ou mais jogos. Após o teste todos os usuários responderam a um questionário com perguntas referente a jogabilidade, entretenimento e aprendizado com relação ao sistema, para levantar dados sobre a eficiência do sistema como substituto dos controles tradicionais. Os participantes

¹ <https://github.com/pauloolileal/joyspeech>

não se igualaram na quantidade de pessoas. Foi possível conseguir somente três pessoas com deficiência nos membros superiores que se encaixavam no perfil, dentre todos os que se candidataram a participaram em contrapartida a dez usuários não deficientes.

Os participantes sem deficiência obtiveram o melhor desempenho, tendo todos completado pelo menos uma fase em cada jogo testado no menor tempo. Já as pessoas com deficiência completaram no mínimo uma fase em pelo menos um dos jogos testados.

Para analisar os resultados, os participantes com deficiência foram agrupados no grupo um e os sem deficiência como grupo dois. Quando questionado sobre a diversão utilizando o sistema 60% do grupo um considerou como muito divertido o grupo dois foram mais de 80%. A respeito da substituição dos controles clássicos ambos consideraram somente 30% para uma substituição total. Em relação a experiência com os jogos o grupo um respondeu que para todos os testados a experiência foi a mesma enquanto que o grupo dois somente 50% mantiveram a resposta. Sobre o conhecimento do jogo escolhido, 60% conhecia pouco e no grupo dois 60% conhecia muito. A respeito da dificuldade de aprendizado o grupo um, 30% considerou fácil, 30% normal e o restante difícil, já o grupo dois 90% considerou como fácil. A respeito do tipo de configuração de comandos ambos os grupos preferiram as específicas do jogo em totalidade. Quanto a dificuldade no grupo um 60% considerou como normal e 90% do grupo dois como fácil. Nos jogos ambos os grupos consideraram o *Sonic Mania* como muito fácil, *Super Mario* como 60% dificuldade média e *Mark of the Ninja* como Muito difícil.

Os resultados mostram que o sistema conseguiu corresponder às expectativas, visto que a grande maioria dos indivíduos confirma que a aprendizagem e usabilidade do sistema é fácil/mediana. As respostas apontam que os jogos testados também influenciaram muito as opiniões dos participantes. Jogos com muitos comandos ou com dificuldade elevada tornaram-se mais difíceis pela jogabilidade, além de a quantidade de comandos confundir o usuário durante o jogo, causando muitos erros durante uma partida.

V. CONCLUSÃO E TRABALHOS FUTUROS

Neste artigo foi apresentada uma solução para permitir que pessoas com deficiência física joguem jogos digitais, de maneira satisfatória e com baixo custo adicional. Os testes, que mesmo com poucos participantes, indicaram ser uma solução válida para os deficientes que mostraram o devido interesse com o sistema e jogos em si. Mesmo que nunca tivessem jogado antes e só conheciam os jogos através de vídeos e por meio de outras pessoas, o entusiasmo foi o suficiente para que mesmo um sistema completamente diferente do tradicional pudesse ser aprendido em minutos. E isso sinaliza como esse mercado é pouco explorado e

que com baixo investimento o número de jogadores pode aumentar rapidamente.

Os participantes que já tinham experiência de longo prazo com jogos relataram que o tempo de processamento é praticamente imperceptível e que para jogos do gênero abordado, plataforma, o comando de voz é eficiente entretanto para outros gêneros os controles tradicionais ainda devem ser mais eficientes.

O sistema desenvolvido apresentou precisão de 70% para o reconhecimento de voz, o que é satisfatório, pois mesmo com o erro de alguns comandos, ainda é uma solução viável, principalmente financeiramente já que o sistema é gratuito (disponível no GitHub como dito na Seção III), o preço de um microfone é baixo (entorno de R\$ 50,00) e as alternativas tem preços muito elevados, como apresentado na Seção IV.

Para trabalhos futuros, propõe-se o desenvolvimento de algoritmos para extração de ruídos em tempo real além da aplicação de inteligência artificial, de modo a melhorar o reconhecimento de comandos aumentando a precisão.

REFERENCES

- [1] S. J. CZAJA and C. C. LEE, “The impact of aging on access to technology,” *Springer-Verlag*, dec 2006.
- [2] C. HADZINSKY, “A look into the industry of video games past, present, and yet to come,” *CMC Senior Theses. Paper 842.*, 2014.
- [3] M. M. Mustaquim, “Automatic speech recognition - an approach for designing inclusive games,” *Springer Science+Business Media*, nov 2011.
- [4] V. B. IRA, “Game on! the ablegamers foundation,” *EP Magazine*, jun 2013.
- [5] INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA (Instituto Brasileiro de Geografia Estatística), “Censo demográfico de 2010,” IBGE. Disponível em: https://biblioteca.ibge.gov.br/visualizacao/periodicos/94/cd_2010_religiao_deficiencia.pdf, 2010.
- [6] F. Fava, “Jogando com o ar: o sopro como instrumento de acessibilidade nos jogos eletrônicos,” *Proceedings of SBGames*, vol. 8, pp. 115–121, 2008.
- [7] S. A. López, F. Corno, and L. D. Russis, “Playable one-switch video games for children with severe motor disabilities based on gnomon,” in *2015 7th International Conference on Intelligent Technologies for Interactive Entertainment (INTETAIN)*, June 2015, pp. 176–185.
- [8] Google. (2018, mar) Speech api. [Online]. Available: <https://cloud.google.com/speech/?hl=pt-br>
- [9] Universidade Federal do Pará (Universidade Federal do Pará). (2018, mar) Fala brasil. [Online]. Available: <https://tobiigaming.com/>
- [10] Microsoft. (2018, mar) Microsoft speech. [Online]. Available: <https://developer.microsoft.com/pt-br/windows/speech>
- [11] M. Noonan. (2017, ago) Windows input simulator. [Online]. Available: <https://archive.codeplex.com/?p=inputsimulator>