

CENTRO UNIVERSITÁRIO SAGRADO CORAÇÃO

LUIS FILIPE GRAEL TINÓS

**GAME DE EXPLORAÇÃO COM APRENDIZADO DE
MÁQUINA E ENGAJAMENTO EM PROJETOS
SOCIAIS**

BAURU
2022

LUIS FILIPE GRAEL TINÓS

**GAME DE EXPLORAÇÃO COM APRENDIZADO DE
MÁQUINA E ENGAJAMENTO EM PROJETOS
SOCIAIS**

Monografia de Iniciação Científica
apresentada a Pró-Reitoria de Pesquisa e
Pós-graduação do curso de Jogos Digitais
do Centro Universitário Sagrado Coração.

Orientador: Prof. M.e Vinicius Santos
Andrade

Co-Orientador: Prof M.e Patrick Pedreira
Silva.

BAURU
2022

RESUMO

Os jogos digitais estão em constante ascensão. É inevitável o impacto dessa tecnologia na vida das pessoas. Além de entretenimento, os jogos passaram a desenvolver um papel mais sério dentro da sociedade. Isso permite a relação dos jogos digitais com diversos outros temas, bem como, atividades e projetos sociais que visam beneficiar uma e/ou um grupo de pessoas. O presente trabalho propõe o uso de técnicas e métodos computacionais para a criação de uma prévia de um jogo de exploração, para dispositivos móveis, com reconhecimento de padrões via redes neurais, ficando o progresso no jogo sujeito a apresentação de imagens (via câmera do dispositivo) para ser processada e reconhecida. Além disso, é de interesse do projeto propiciar ferramenta que suporte o estímulo a realização de ações em benefício a instituições/entidades cadastradas em troca de recompensas e do próprio progresso na narrativa do jogo. Para o cumprimento da proposta, foram investigados, por meio de treinamento e validações, modelos de redes neurais convolucionais para a classificação das imagens apresentadas. A relação entre jogador e entidade/instituição foi investigada e implementada para uma futura proposta, essa, capaz de perdurar e ainda se mostrar cativante, no objetivo de atrair o maior número de jogadores possível, ao mesmo tempo em que o projeto vai sendo expandido. A pesquisa mostrou um resultado muito satisfatório, comprovando a possibilidade de uso das redes neurais convolucionais em um jogo mobile, com ótimas métricas e, ainda, de forma otimizada, proporcionando uma experiência gratificante para o usuário.

Palavras-chave: Jogos Digitais; Aprendizado de Máquina; Gamificação; Visualização Computacional.

ABSTRACT

Digital games are constantly rising. The impact of this technology on people's lives is inevitable. In addition to entertainment, games began to develop a more serious role within society. This allows the relationship of digital games with several other themes, as well as social activities and projects that aim to benefit one and/or a group of people. The present work proposes the use of techniques and computational methods for the creation of a preview of an exploration game, for mobile devices, with pattern recognition via neural networks, with the progress in the game subject to the presentation of images (via the device's camera) to be processed and recognized. In addition, it is in the interest of the project to provide a tool that supports the stimulus to carry out actions for the benefit of registered institutions/entities in exchange for rewards and progress in the game's narrative. In order to fulfill the proposal, models of convolutional neural networks were investigated, through training and validation, to classify the images presented. The relationship between player and entity/institution was investigated and implemented for a future proposal, this one, capable of lasting and still be captivating, in order to attract as many players as possible, while the project is being expanded. The research showed a very satisfactory result, proving the possibility of using convolutional neural networks in a mobile game, with great metrics and, also, in an optimized way, providing a rewarding experience for the user.

Keywords: Digital Games; Machine Learning; Gamification; Computacional Visualization.

LISTA DE ILUSTRAÇÕES

1.	INTRODUÇÃO	5
1.1	HISTÓRIA DOS JOGOS	5
1.2	INTELIGÊNCIA ARTIFICIAL	7
1.3	DISPOSITIVOS MÓVEIS	13
1.4	O ROTEIRO E OS JOGOS DE EXPLORAÇÃO	13
2.	OBJETIVOS	16
2.1	OBJETIVO GERAL	16
2.2	OBJETIVOS ESPECÍFICOS	16
3.	JUSTIFICATIVA	17
4.	METODOLOGIA	19
4.1	BANCO DE IMAGENS	19
4.2	VISÃO COMPUTACIONAL	21
4.3	DESENVOLVIMENTO DO JOGO	24
4.4	ARQUITETURA DAS APLICAÇÕES	31
4.5	ENGAJAMENTO SOCIAL	32
5.	RESULTADOS OBTIDOS	33
6.	DISCUSSÃO DOS RESULTADOS	40
7.	CONSIDERAÇÕES FINAIS	41
8.	ORÇAMENTO	42
	REFERÊNCIAS	43
	ANEXO A – GAME DESIGN DOCUMENT – SOMEBODY	48

Dados Internacionais de Catalogação na Publicação (CIP)
de acordo com ISBD

Tinós, Luís Filipe Grael

T587g

Game de exploração com aprendizado de máquina e
engajamento em projetos sociais / Luís Filipe Grael
Tinós. -- 2022.

60f. : il.

Orientador: Prof. M.e Vinicius Santos Andrade

Coorientador: Prof. M.e Patrick Pedreira Silva

Monografia (Iniciação Científica em Jogos Digitais) -
Centro Universitário Sagrado Coração - UNISAGRADO
- Bauru - SP

1. Jogos Digitais. 2. Aprendizado de Máquina. 3.

Elaborado por Lidyane Silva Lima - CRB-8/9602

SUMÁRIO

1. INTRODUÇÃO	5
1.1 HISTÓRIA DOS JOGOS	5
1.2 INTELIGÊNCIA ARTIFICIAL	7
1.3 DISPOSITIVOS MÓVEIS	13
1.4 O ROTEIRO E OS JOGOS DE EXPLORAÇÃO	13
2. OBJETIVOS	16
2.1 OBJETIVO GERAL	16
2.2 OBJETIVOS ESPECÍFICOS	16
3. JUSTIFICATIVA	17
4. METODOLOGIA	19
4.1 BANCO DE IMAGENS	19
4.2 VISÃO COMPUTACIONAL	21
4.3 DESENVOLVIMENTO DO JOGO	24
4.4 ARQUITETURA DAS APLICAÇÕES	31
4.5 ENGAJAMENTO SOCIAL	32
5. RESULTADOS OBTIDOS	33
6. DISCUSSÃO DOS RESULTADOS	40
7. CONSIDERAÇÕES FINAIS	41
8. ORÇAMENTO	42
REFERÊNCIAS	43
ANEXO A – GAME DESIGN DOCUMENT – SOMEBODY	48

1. INTRODUÇÃO

É fato que os jogos digitais desempenham um papel extraordinário em relação a entretenimento, ensino, aprendizado e até mesmo causas sociais, na atualidade. É uma atividade que ganhou destaque ao longo dos anos e se encontra presente nos videogames, computadores e dispositivos móveis.

Também é fato que os *smartphones* e os *tablets* se tornaram aparelhos indispensáveis para muitas pessoas, fazendo parte do cotidiano da sociedade, com cerca de 424 milhões de dispositivos ativos só no Brasil. (FGV, 2020).

Além disso, a Inteligência Artificial também cresceu. Prova disso é o grande uso e poder de resolução que as assistentes pessoais dos dispositivos móveis possuem. Tais técnicas também podem ser direcionadas para alavancar o desenvolvimento de soluções gamificadas em prol do lúdico e da responsabilidade social.

Considerando tudo o que foi exposto, o presente projeto tem o intuito de mostrar, a partir de uma prévia (*preview*) de um jogo para dispositivos móveis que utilize o reconhecimento de objetos do mundo real para interação digital no roteiro do game, como a utilização dessas tecnologias são válidas e funcionam até para os dispositivos mais antigos, atrelado ao engajamento social, ambos como recursos que viabilizam o progresso no game.

A seguir, um curto referencial teórico é exposto no intuito de fundamentar esse projeto de pesquisa.

1.1 HISTÓRIA DOS JOGOS

O surgimento dos jogos se deu há muito tempo, mas sem data definida. (CARVALHO, 2018). Existem indícios de que em algumas partes da Ásia, Austrália, América pré-colombiana e alguns povos indígenas das ilhas mais remotas do Oceano Pacífico eram responsáveis pelo desenvolvimento de jogos que, no contexto da época, eram utilizados para a satisfação recreativa, práticas religiosas, e até mesmo, como forma de expressão, sendo benéficos em vários sentidos (CARVALHO, 2018 apud RAMOS, 1982) e, então, culminando na evolução desses jogos e o surgimento dos primeiros jogos olímpicos.

Os jogos podem ser definidos como uma atividade de aprendizado, uma ação que ultrapassa os campos físicos: é uma ação voluntária representando uma fuga ao estado real (LUCHESE, RIBEIRO; 2009 apud HUIZINGA, 2003). Também pode se dizer que os jogos são uma ferramenta fundamental de desenvolvimento de habilidades, tais como: comunicação, sociabilidade e aperfeiçoamento do raciocínio lógico (CARVALHO, 2018), sendo ferramentas de instrução e exercício da mente e intelecto (TAROUCO *et al.*, 2004). Nesse sentido, a definição de jogos é muito ampla, já que é possível utilizar a palavra “jogo” para inúmeras ações: jogar, brincar, divertir-se; jogo, também, pode ser uma competição, um desafio (SILVA *et al.*, 2009).

Jogar é um componente de jogo. A experiência de jogar é apenas uma das muitas maneiras de olhar e entender jogos, entender jogar o jogo representa um aspecto do jogo. Jogar seria um elemento crucial para um largo conceito de jogos. Jogar é um subconjunto de jogos. (SILVA *et al.*, 2009, p. 4).

Os jogos possuem algumas características marcantes: a representação, a interação, o conflito e a segurança (LUCHESE, RIBEIRO, 2009 apud CRAWFORD, 1982). A interação é capaz de mudar a realidade apresentada a partir de escolhas e modos de jogar. O conflito está relacionado a um problema e/ou um obstáculo que pode ter sido gerado pela própria interação e a segurança é proporcionada aos usuários a fim de não estabelecer riscos físicos a um indivíduo durante um conflito no universo fictício (LUCHESE, RIBEIRO, 2009 apud CRAWFORD, 1982). É possível deduzir que todas essas características estão diretamente ligadas e são complementares, umas às outras.

Os jogos eletrônicos, ou jogos digitais, tiveram seu surgimento nos anos 50 e foram criados a partir de estratégias militares em computadores e testes de inteligência artificial (CARVALHO, 2018).

Com o passar dos anos, as tecnologias utilizadas na criação dos games se tornaram cada vez mais avançadas, o que possibilitou a criação de jogos mais desenvolvidos e detalhados. Empresas como Nintendo, Sega, Sony, surgiram no mercado dos consoles (BATISTA *et al.*, 2007) e venderam inúmeras unidades de jogos. O crescimento tecnológico possibilitou, também, a presença dos jogos digitais nos computadores (CARVALHO, 2018).

Levando em consideração o avanço do jogo como atividade, é possível afirmar que os jogos passaram a ser vistos de forma mais profunda. Já não se tem mais a

ideia de um jogo como forma de entretenimento. Mas, também, como forma de aprendizado e sua utilização no ensino (CARVALHO, 2018).

Os jogos digitais cresceram e continuam crescendo. Nos dias atuais, é possível integrar vários tipos de tecnologia a eles. Pode-se destacar o uso da inteligência artificial nos jogos. Esse tipo de tecnologia está sendo explorada em *games* e tem se mostrado muito eficiente e interessante nesse contexto.

1.2 INTELIGÊNCIA ARTIFICIAL

O termo Inteligência Artificial (IA) surge da ideia de o ser humano deixar as máquinas inteligentes. Nessa linha de pesquisa, os computadores adquirem habilidades que simulam a inteligência dos humanos. (NAKABAYASHI, 2009). A inteligência artificial estuda essa emulação comportamental (NAKABAYASHI, 2009 apud MCARTHY, 1956). As habilidades são desenvolvidas a partir de uma inteligência, para as máquinas, criada pelos humanos (NAKABAYASHI, 2009).

A inteligência artificial representa um papel muito importante no cotidiano. Ela está presente em sistemas projetados para a resolução de problemas em relação à economia; com atividades que beneficiam a sociedade como um todo e, também, tem grande participação na pesquisa científica e tecnológica (CERRI, CARVALHO; 2017). Uma ramificação da inteligência artificial é o aprendizado de máquina (*machine learning*), que permite ser conceituado, de forma ampla, como métodos computacionais que usam experiência (tipicamente dados coletados) no intuito de aprimorar performance ou fazer previsões com exatidão (MOHRI, ROSTAMIZADEH, TAWALKAR; 2018).

Isso leva ao desenho de programas capazes de aprender por si sós, utilizando-se um conjunto de dados que representam experiências passadas. Trata-se de uma área de pesquisa multidisciplinar que engloba inteligência artificial, probabilidade e estatística, teoria da complexidade computacional, teoria da informação, filosofia, psicologia, neurobiologia, entre outros. (CERRI; CARVALHO, 2017, p. 298).

Desse modo, é possível relacionar a técnica de aprendizado de máquina com a própria obtenção de conhecimento do ser humano. A técnica ainda é utilizada para reconhecer toda a biologia de uma flor ou planta da mesma espécie, mas de tipos diferentes, por exemplo. O algoritmo vai analisar as características das flores/plantas

e, por meio de treinamento, categorizar as flores/plantas diferentes (CERRI, CARVALHO, 2017).

Tom Mitchell (2006) elenca que o *machine learning* tem sucesso em aplicações no mundo real, tais como o reconhecimento de voz, visão computacional, controle robótico etc.

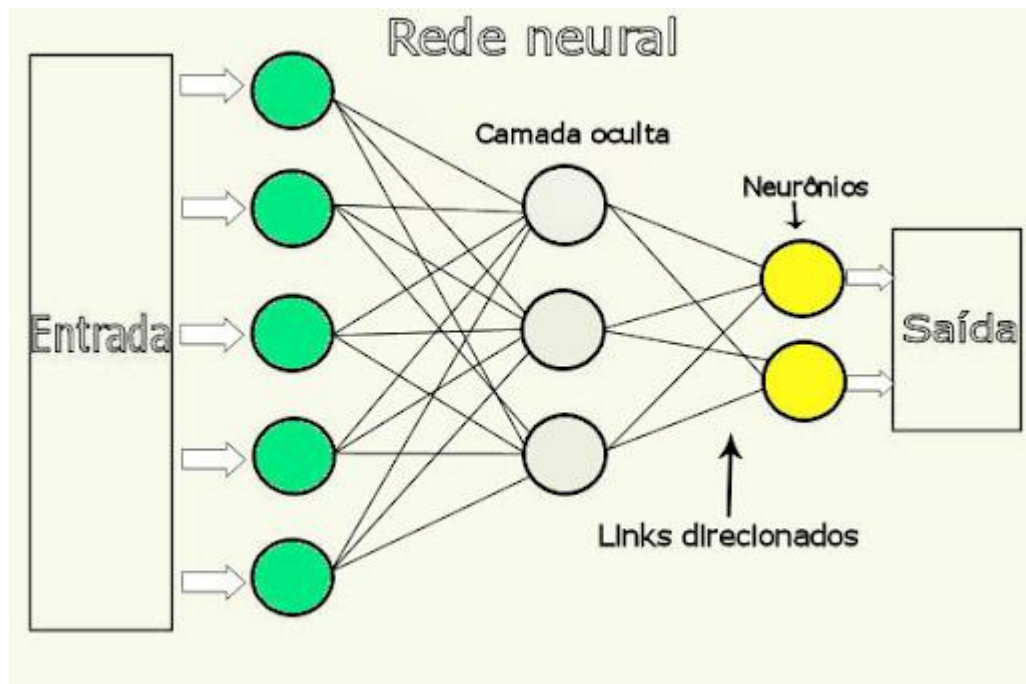
As redes neurais artificiais são outro modo de resolver problemas dentro do campo da inteligência artificial (BARRETO, 2002). Em alternativa à criação de um algoritmo que vai jogar xadrez, por exemplo, simulando um aluno, é possível desenvolver uma rede de circuitos neurais, a partir de um circuito computacional. Com treino e tempo, o melhor resultado esperado é o desenvolvimento do aprendizado, seja com tentativa e erro, com acertos, com o aprendizado de novas jogadas, podendo até ultrapassar o próprio professor (BARRETO, 2002). Desse modo, com o desenvolvimento das redes neurais, é possível detectar comportamentos imprevisíveis dos circuitos neurais quando apresentados em ambientes diversos e, até mesmo, obter deles um comportamento que não pode ser previsto e/ou compreendido (BARRETO, 2002).

Pode-se dizer que as redes neurais têm a capacidade de resolver problemas extremamente complexos através das suas características. Uma característica marcante é seu poder de generalização, ou seja, de imprimir um resultado que não foi previamente apresentado aos circuitos neurais artificiais durante o período de treinamento, mas, que, após reconhecer características similares, produziu a saída adequada (HAYKIN, 2007). Recurso esse fortemente necessário para a proposta desse projeto, no que tange o reconhecimento de objetos comuns e amplamente conhecidos no mundo real.

Especificamente, um problema complexo de interesse é decomposto em um número de tarefas relativamente simples, e atribui-se a redes neurais um subconjunto de tarefas que coincidem com as capacidades inerentes. (HAYKIN, 2007, p. 29).

A Figura 1 apresenta uma exemplificação da representação de uma rede neural Perceptron de múltiplas camadas (MLP).

Figura 1 - Forma de representar uma rede neural artificial

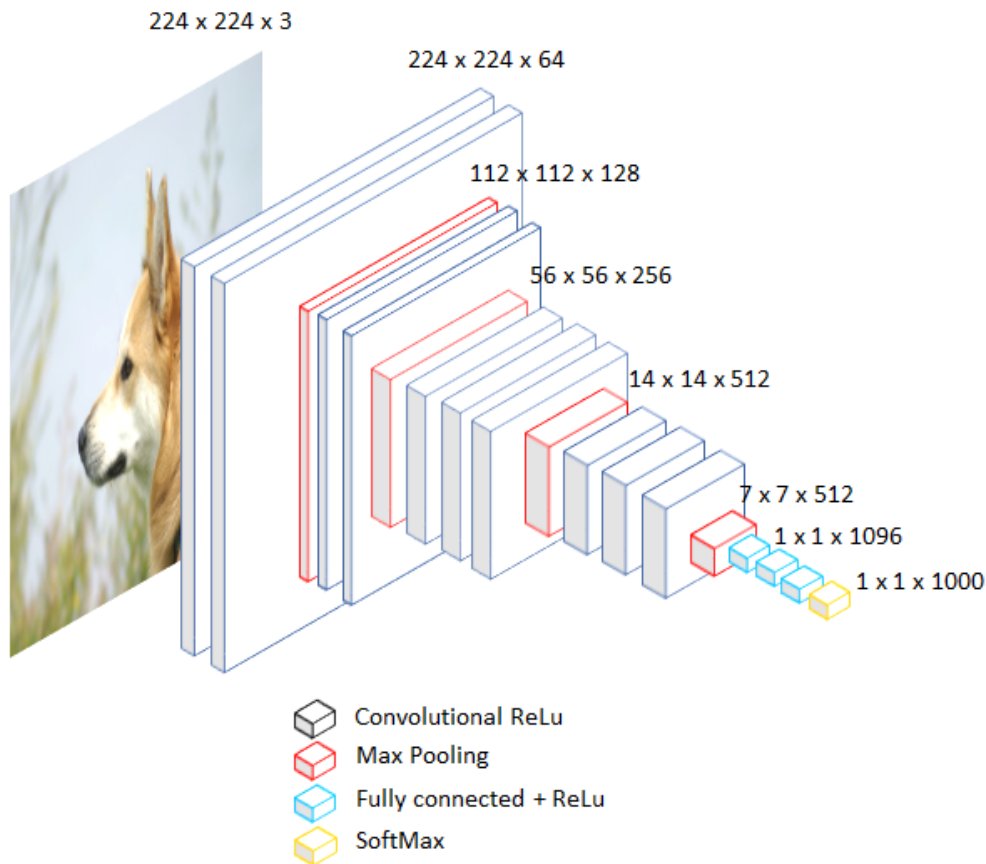


Fonte: PETNews (2011).

Dentre os tipos de algoritmos de *machine learning* e de redes neurais, existem as redes neurais convolucionais (CNN), utilizadas no presente trabalho. Esse algoritmo realiza a classificação das imagens a partir de convoluções, que resultam em uma matriz de números para que o algoritmo possa identificar o(s) objeto(s) apresentado(s) de forma a correlacioná-los com a classe correta.

A Figura 2 representa a arquitetura de uma rede neural convolucional e suas camadas.

Figura 2 – Exemplo da arquitetura de uma rede neural convolucional



Fonte: Towards Data Science (2020).

A imagem acima demonstra a arquitetura de uma rede neural convolucional. Existem três tipos de camadas e subdivisões delas.

Camada de entrada (*input layer*): é a camada responsável por receber a imagem a ser classificada pela rede. Em uma rede neural convolucional, é possível inserir imagens como um todo em sua camada de entrada – e não apenas metadados. Isso é vantajoso, já que é possível trabalhar com etapas de pré-processamento e processamento de imagens dentro do próprio algoritmo.

Camada oculta (*hidden layer*): essa é a camada onde ocorrem as convoluções, os *poolings* e, também, é onde a classificação é realizada. Dentro da camada oculta, são definidos os pesos e vieses dos parâmetros relacionados à imagem e/ou grupo de imagens utilizadas para o treinamento da rede. Esses parâmetros são obtidos pela rede ao treinar o modelo com uma base de dados rotulada (com *labels*). Subdividindo essa camada, obtemos a Camada Convolucional (*convolutional layer*): responsável por convolver a imagem a partir de uma matriz chamada kernel – ou janela, filtro –

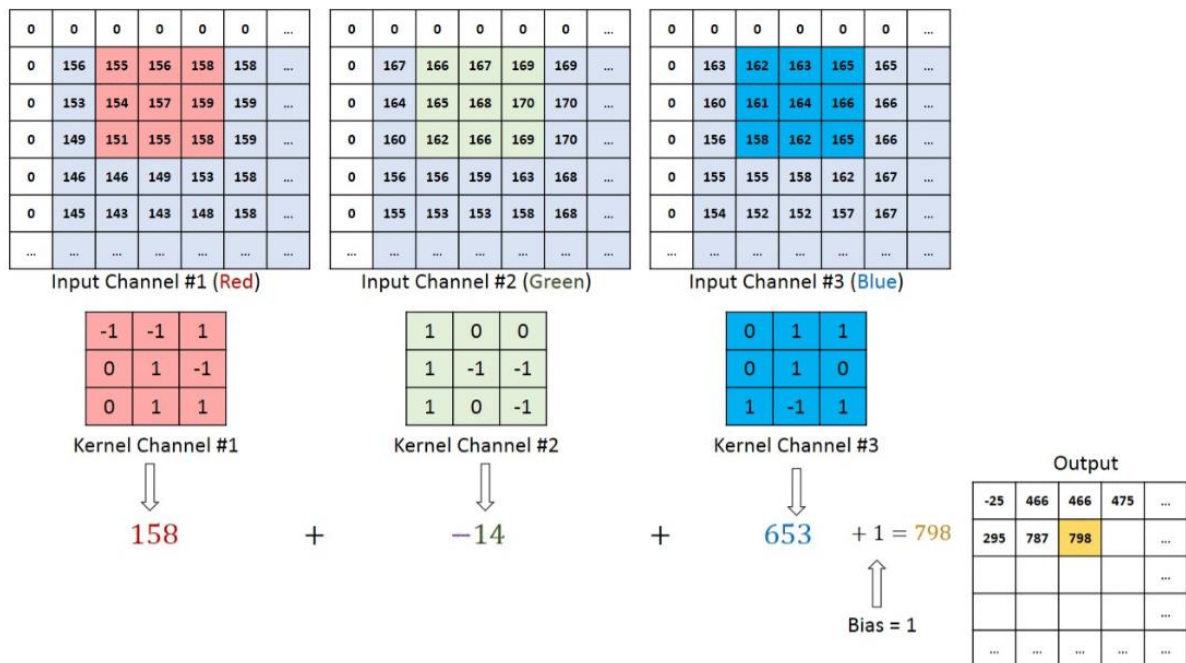
que retorna uma matriz numérica com dimensões iguais às da imagem de entrada. Essa matriz é o resultado de várias convoluções, em que, em cada uma delas, é multiplicado o valor do peso do parâmetro pelo valor do atributo do pixel alinhado à posição do pixel do kernel e feita a soma dos valores resultantes. Essa soma é um dos valores que formam a matriz final.

$$R = w_1x_1 + w_2x_2 \dots w_nx_n = \sum_{i=1}^n w_ix_i \tag{1}$$

A fórmula acima representa uma convolução em uma rede neural, em que w é o valor de cada peso, x o valor de cada atributo e R o valor resultante que forma a matriz final.

A Figura 3 mostra como as convoluções agem em uma rede neural. As matrizes 3x3 (vermelha, verde e azul) representam os kernels de cada canal de cor RGB da imagem.

Figura 3 - Representação de convoluções em uma rede neural convolucional



Fonte: Neuronio BR (2018).

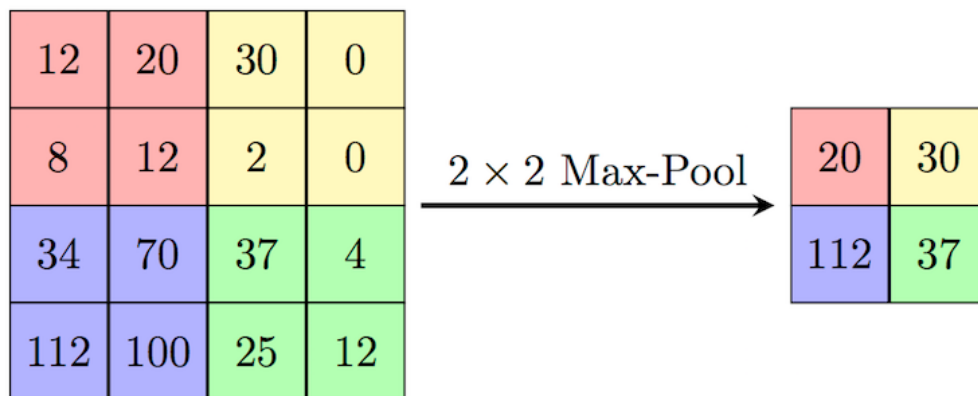
Desse modo, a matriz numérica resultante irá extrair informações relevantes para a classificação da imagem de entrada. Durante o treinamento da rede, a cada processamento, os kernels sofrem ajustes e são acionados apenas quando a entrada

tiver uma informação característica e padronizada ou comum às outras imagens utilizadas no treinamento do algoritmo (RODRIGUES, 2018).

A Camada de *Pooling* (*pooling layer*): responsável por reduzir o tamanho dos dados obtidos a partir da imagem de entrada. Ela, normalmente, é seguida de uma camada convolucional e é importante para o benefício da rede, ao tempo em que reduz o processamento e o custo de memória, já que o tamanho da matriz enviada à próxima camada é menor (RODRIGUES, 2018).

A Figura 4 exemplifica o procedimento de *pooling* com a técnica de *max-pooling*, onde a matriz resultante é o maior valor de cada um dos quatro conjuntos de dados.

Figura 4 - Exemplo da aplicação de *max-pooling*



Fonte: Computers Science Wiki (2018).

A Camada Totalmente Conectada (*fully connected layer*): encontram-se, geralmente, no final da rede. Elas enviam a resposta da classificação para a(s) camada(s) de saída da rede (RODRIGUES, 2018). Essas camadas utilizam as características mais importantes extraídas das imagens nas camadas anteriores para retornar uma saída – ou seja, a qual classe pertence a suposta entrada.

As redes neurais convolucionais – que serão ainda mais exploradas na metodologia do presente trabalho – conseguem receber imagens na camada de entrada e, além de classificar imagens, podem ser usadas para a resolução de diversos outros problemas, como o reconhecimento de áudios e objetos (SILVA, 2018).

Desse modo, a inteligência artificial é fortemente válida para a construção de *games* e a visão computacional (extração de dados de imagens e impressão de

informações a partir desses dados) pode ser muito bem implementada em celulares, devido à mobilidade e portabilidade dos mesmos.

1.3 DISPOSITIVOS MÓVEIS

É evidente que os dispositivos móveis influenciaram, dinamizaram e continuam alterando a forma em que as pessoas vivem. A telecomunicação sempre esteve presente no cotidiano da população. Em 1947, surgiu um sistema de utilização de telefonia móvel, criado pela empresa americana Bell Company, utilizando o conceito de áreas de cobertura. Porém, esse mesmo sistema foi considerado inviável na época. Somente em 1973 que a tecnologia foi incorporada a um aparelho celular de uso pessoal, pela Motorola (ABREU, 2005).

A tecnologia dos dispositivos móveis evoluiu culminando no que atualmente é conhecido como *smartphones* e *tablets* (HIGUCHI, 2011), com características e funcionalidades múltiplas, complexas e atraentes.

Assim, os celulares e/ou *smartphones* constituem um papel importante na evolução da tecnologia e no comportamento social. É possível chegar em diferentes locais a partir dele (com os softwares de localização e transporte), pedir comida, conversar por mensagens, por chamadas de vídeo, jogar, planejar eventos. Antes, na primeira geração, tínhamos apenas os serviços de voz (MATOS *et al.*, 2019). Hoje, temos uma infinidade de serviços.

1.4 O ROTEIRO E OS JOGOS DE EXPLORAÇÃO

Um dos primeiros passos do desenvolvimento de um *game* se baseia na criação do roteiro, em que serão descritas a narrativa, a relação dos itens, as ações sociais; tudo com base na jornada do herói. Essa jornada, definida como monomito, por Campbell (1997), pode ser considerada como uma série de acontecimentos que envolvem o herói, desde sua partida, o chamado para a aventura, até o retorno, a liberdade para viver (CAMPBELL, 1997).

A jornada do herói é uma estrutura narrativa que foi construída a partir da análise de mitos e contos diferentes por Joseph Campbell (MARTINEZ, 2008). Nos enredos analisados, o autor identificou a existência de características semelhantes

que separam as narrativas em partes: a partida, a iniciação e o retorno (MARTINEZ, 2008).

No entanto, a estrutura narrativa não serve apenas para contos e mitos. Os passos da jornada do herói podem ser desenvolvidos em qualquer história. (VOGLER, 2006).

A beleza do modelo da Jornada do Herói é que não apenas descreve um padrão nos mitos e contos de fadas, mas também é um mapa muito exato do território que se deve percorrer para ser um escritor. Ou, aliás, um ser humano (VOGLER, 2006).

A Figura 5 apresenta os 12 passos da jornada do herói, utilizada na narrativa do trabalho.

Figura 5 - Os 12 passos da jornada do herói



Fonte: Tolkienista (2020).

No quesito de gêneros, o de exploração é bastante conhecido e faz sucesso no mundo dos *games*. Exemplos de jogos de exploração são o *Zelda: Breath Of The Wild*, que fornece ao jogador um mundo aberto para ser desbravado, explorando a não-linearidade de jogabilidade, e o *Pokemon GO*, já citado anteriormente, que utiliza a câmera e o GPS do celular, aliados à realidade aumentada, para proporcionar

experiências únicas de exploração, como se “trouxessem” os pokemons para o mundo real.

A experiência de exploração e o prazer da descoberta não estão unicamente vinculados ao mundo de jogo, pois também podem ocorrer no espaço de possibilidades ou ações do jogador. A partir de estratégias de tentativa e erro, o jogador pode combinar elementos do jogo, descobrindo nessa dinâmica novos resultados. (NAKAMURA, 2013, p. 24).

Dessa forma, os jogos de exploração podem proporcionar experiências diferentes para cada jogador, o que possibilita a interação deles para a discussão de possibilidades de resolução de enigmas e/ou para o compartilhamento das experiências de cada um deles.

2. OBJETIVOS

A seguir são apresentados os objetivos que pautam o desenvolvimento desse projeto.

2.1 OBJETIVO GERAL

Estudar e demonstrar a viabilidade da produção de jogo digital para o engajamento social e a progressão pautada em técnicas de aprendizado de máquina.

2.2 OBJETIVOS ESPECÍFICOS

- a) Levantar referencial teórico e bibliográfico a respeito das técnicas que serão utilizadas para o desenvolvimento do projeto;
- b) Apresentar um roteiro inovador e adequado a proposta de desenvolvimento do jogo;
- c) Selecionar e constituir um banco de imagens próprio com objetos juntamente da realização de procedimentos de processamento de imagens;
- d) Estabelecer modelos de reconhecimento de padrões por redes neurais artificiais para a identificação de objetos determinados em imagens;
- e) Desenvolver protótipo de jogo digital que implementa as técnicas e modelos definidos e abordagem de engajamento social;
- f) Analisar resultados das técnicas empregadas, do protótipo e a viabilidade de expansão;
- g) Promoção do projeto e discussão da abordagem utilizada em eventos relacionados à área acadêmica.

3. JUSTIFICATIVA

Tecnologias diferentes têm se misturado para projetar e executar softwares e aplicações complexas com o intuito de promover experiências diversas e satisfatórias aos usuários.

Dentre as várias tecnologias presentes nos jogos, a inteligência artificial mostra-se muito utilizada. Um exemplo do uso da IA nos jogos são as máquinas de estado finitas, que ocasionam disparos de estados a cada interação ocorrida com o objeto ou agente ao qual a máquina está relacionada.

Quando unimos esses dois mundos, da IA com os jogos, apresentamos uma grande área de pesquisa que vem aumentando consideravelmente o seu nível de importância na comunidade científica e na indústria. Por um lado, o interesse fundamental dos pesquisadores em IA pelos jogos vem do fato que eles se mostram como um ótimo laboratório de experimentação para pôr em prática e avaliar as técnicas desenvolvidas. Por outro lado, as técnicas de IA permitem aumentar o grau de realismo dos jogos, assim como o grau de imersão dos jogadores, tornando a experiência do ato de jogar cada vez mais agradável. Isso pode ser constatado através de inúmeras pesquisas conforme publicações efetuadas em grandes conferências da área tais como a *IEEE Conference on Computational Intelligence and Games* e a *AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (ARANHA; ARAÚJO; MADEIRA, 2018, p. 2).

Uma técnica que é amplamente usada em aplicações – principalmente para dispositivos móveis – é a de classificação de imagens com o uso de *machine learning*.

Pelo fato de que os *smartphones* – em sua grande maioria – possuem uma câmera nativa do dispositivo e algoritmos, frameworks e bibliotecas que dão suporte para o uso da IA nessas máquinas, a aplicação desse método para a criação de sistemas voltados para a classificação de imagens é bastante utilizada. A classificação de imagens de sementes (PIRES *et al.*, 2020) e a classificação de obras artísticas (SOUZA; SOARES FLORES, 2020) são alguns exemplos do uso da classificação de imagens em aplicações.

Porém, apesar de a classificação de imagens a partir de fotos de objetos ser uma técnica muito empregada em muitos tipos de softwares, existem poucos jogos que utilizam esse tipo de tecnologia com esse fim, ainda mais relacionada ao avanço linear narrativo.

Além disso, atualmente, muitas pessoas não conseguem e/ou não tem tempo de focar em alguma atividade beneficente. As pessoas até desejam ajudar de alguma forma, porém, em muitas vezes, não conseguem se programar para esse tipo de ação.

Os projetos sociais são muito importantes para o desenvolvimento de uma sociedade em si. Cada ato beneficente e cada iniciativa contribuem para a reformulação da sociedade.

Os projetos sociais nascem do desejo de mudar uma realidade. Os projetos são pontes entre o desejo e a realidade. São ações estruturadas e intencionais, de um grupo ou organização social, que partem da reflexão e do diagnóstico sobre uma determinada problemática e buscam contribuir, em alguma medida, para “um outro mundo possível” (STEPHANOU, MÜLLER, CARVALHO; 2003, p. 11).

Uma maneira de aproximar o ser humano de entidades/instituições com projetos sociais é o uso de mecânicas que sugerem um desafio. Esse tipo de mecânica estimula a sociedade a participar e contribuir com os as entidades/instituições. Um exemplo de desafio é “desafio do balde de gelo”, ou “*ice bucket challenge*”, de 2014, que arrecadou fundos que serviram para pesquisas na luta contra a esclerose lateral amiotrófica (ALS ASSOCIATION, 2019).

Outras atividades são importantes na luta a favor de causas sociais, como o apoio à organização “*Black Lives Matter*”, que teve início em 2013.

O Black Lives Matter, às vezes citado nos cartazes como BLM, é uma organização que nasceu em 2013 por três ativistas norte-americanas: Alicia Garza, da aliança nacional de trabalhadoras domésticas; Patrisse Cullors, da coalizão contra a violência policial em Los Angeles; e Opal Tometi, da aliança negra pela imigração justa. Hoje, é uma fundação global cuja missão é “erradicar a supremacia branca e construir poder local para intervir na violência infligida às comunidades negras” pelo Estado e pela polícia.” (UOL, 2020).

Desse modo, a justificativa da criação do presente trabalho está pautada na importância e necessidade de mostrar a viabilidade do uso de tecnologias como as redes neurais convolucionais para a classificação de imagens em um jogo mobile de forma acessível, buscando, sempre, trazer experiências de jogabilidade inovadoras e diversificadas para o usuário. Ainda, a justificativa se pauta na necessidade de analisar como as técnicas e métodos aplicados podem viabilizar a integração das pessoas aos projetos e ações sociais, apesar de não ser o foco do projeto em questão.

4. METODOLOGIA

O método utilizado nesta pesquisa foi do tipo de análise exploratória, com abordagem quantitativa quando a verificação dos resultados referentes à probabilidade de acertos do modelo inteligente, e qualitativa, com testes de protótipo que serão feitos pelo próprio desenvolvedor, para o estudo de engajamento de ação social vinculada com o jogo proposto.

É importante ressaltar que o produto da pesquisa (um *game*) não foi desenvolvido por completo. O trabalho objetiva analisar a viabilidade de construção desse produto com os métodos propostos em consonância com os objetivos anteriormente estabelecidos.

4.1 BANCO DE IMAGENS

Para treinamento do modelo e generalização do reconhecimento de padrões, a formação e obtenção das informações requisitadas pelas técnicas empregadas no projeto foram adquiridas a partir do próprio desenvolvedor do presente projeto. Para a aquisição dessas imagens, foi utilizada a câmera de celulares (iPhone 6s e iPhone 8) e as imagens foram tiradas nos modos retrato e paisagem. Essas imagens, ou *datasets* – bancos de imagens, no caso -, foram capturadas de várias formas, levando em consideração situações reais em que o usuário tiraria as fotos e apresentaria ao software para posterior classificação: com baixa luminosidade, alta luminosidade (extremos), com sombras, tremidas etc. Assim, o modelo tem uma maior acurácia ao classificar a imagem, já que os exemplos passados a ele, sobre determinada classe (objeto fotografado), são diversos.

Ao todo, foram adquiridas mil e dezessete imagens. Desse total, setecentos e setenta e uma imagens foram utilizadas para treinamento e duzentos e quarenta e seis foram para a validação do modelo de rede neural convolucional.

Por ter uma natureza de aprendizado supervisionado, as *features* – imagens inseridas na camada de entrada - são classificadas em classes já esperadas e as imagens também foram categorizadas em classes diferentes.

A Figura 6 apresenta a quantidade de imagens e suas classes separadas em “*Pens*” (canetas), “*Keys*” (chaves) e “*Buckets*” (baldes).

Figura 6 - Quantidades de imagens para as respectivas
classes e datasets

```
Train Pens: 292  
Train Keys: 292  
Train Buckets: 187  
Validation Pens: 103  
Validation Keys: 103  
Validation Buckets: 40
```

Fonte: Autoria própria.

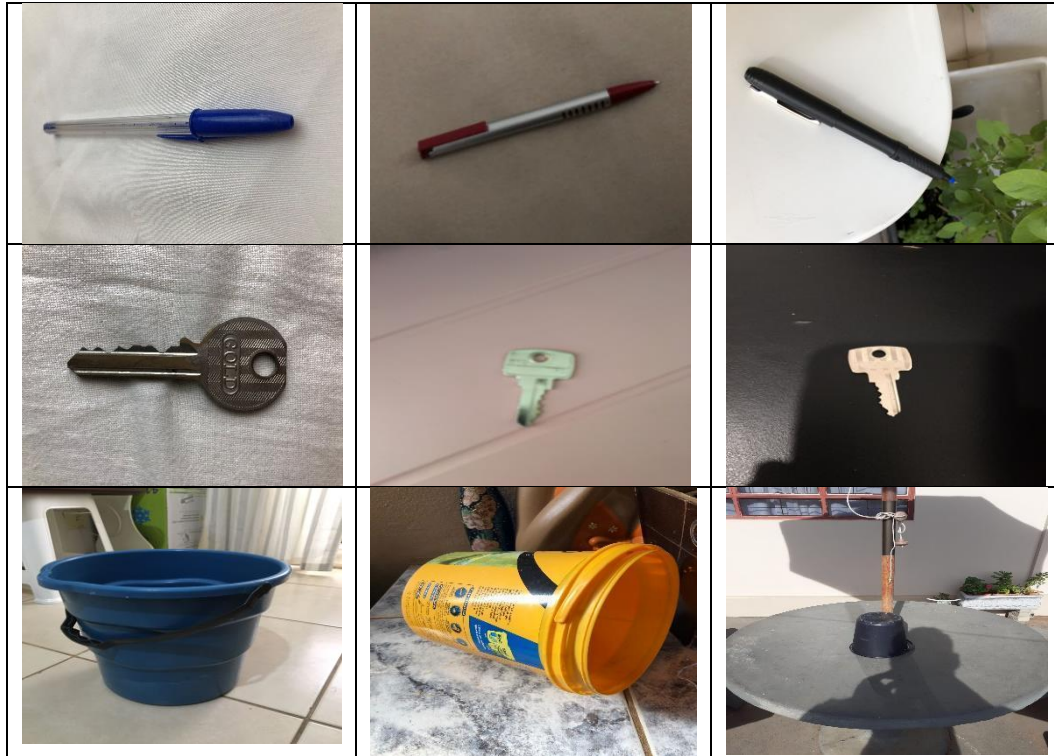
Dentro do *dataset* de treinamento, foram inseridas duzentos e noventa e duas imagens para canetas, duzentos e noventa e duas imagens para chaves e cento e oitenta e sete imagens para baldes. Da mesma forma, no *dataset* de validação, estão presentes cento e três imagens de canetas, cento e três imagens de chaves e quarenta imagens de baldes.

Todas as imagens foram capturadas a partir de dispositivos móveis (iPhone 6S e iPhone 8) buscando utilizar exemplares com características variadas para avaliar corretamente a grande maioria das *features* apresentadas pelos usuários.

O motivo da escolha dos 3 objetos (caneta, chave e balde) está diretamente relacionado com a narrativa do jogo. Ao decorrer da história, o jogador precisará de imagens dos respectivos objetos para a progressão. A caneta foi escolhida para comunicar-se com um senhor com deficiência auditiva; a chave foi escolhida para abrir uma porta trancada na cidade e o balde é necessário para o personagem conseguir pegar água de um poço. Quando o objeto for classificado, o jogador poderá (ou não, caso a classificação seja negativa) progredir na história.

A Figura 7 mostra exemplos de imagens utilizadas para o treinamento da rede neural convolucional.

Figura 7 - Exemplos de imagens de canetas, chaves e baldes que foram utilizadas no treinamento da CNN



Fonte: Autoria própria.

4.2 VISÃO COMPUTACIONAL

A proposta do projeto correlaciona um jogo em ambiente digital com sua progressão interrompida à espera de objetos do mundo real. Sendo assim, a câmera do dispositivo móvel será ativada, o objeto requisitado pelo *game* deverá ser apresentado. O software identifica, realiza o processamento da imagem capturada e a classifica.

O pré-processamento das imagens dos *datasets* de treino e validação foi feito a partir da biblioteca Keras¹ pela facilidade e possibilidade de implementação dentro do algoritmo de criação e treinamento do modelo de *machine learning*. O procedimento de pré-processamento depende muito de como são as imagens dos *datasets*, é muito importante para garantir uma boa performance do algoritmo (caso seja feito um *rescaling* nas imagens da camada de entrada, por exemplo) e, também, tem a tarefa de melhorar e “preparar” a imagem para a classificação.

¹ Framework de código aberto para *Deep Learning*. Disponível em: < <https://keras.io/> >

Durante o pré-processamento, foi feito um *rescaling* (reescalonamento) das imagens para 160x160 pixels. Essa padronização é importante para que as imagens que entram na rede tenham um peso menor e se padronizem em um único tamanho, facilitando o treinamento. Além disso, foi feita a normalização de valores para os pixels da imagem, que estavam em escala RGB de três canais. A normalização é o processo que limita os valores dos pixels da imagem de 0 – 255 para 0 – 1. Toda imagem que entra na rede é pré-processada dessa forma. Assim, a classificação de novas *features* se mantém padronizada e eficiente, visto que esse procedimento de pré-processamento reduz o tamanho da imagem e agiliza o processo de classificação.

Após o pré-processamento, as camadas convolucionais extraem as características – ou atributos – das imagens para que, posteriormente, sejam realizados o processamento e a classificação das mesmas (OLIVEIRA *et al.*, 2019).

Para a etapa de inteligência artificial no uso de aprendizado de máquina para reconhecimento de padrões, foi estudado e implementado o modelo de rede neural convolucional para classificação multiclasse com algoritmo de *backpropagation*, aliado a técnicas como *transfer learning*, *data augmentation* e *early stopping*. O modelo é responsável por classificar as imagens em 3 classes diferentes (caneta, chave ou balde) utilizando *dense layers* com função de ativação *softmax*, retornando a probabilidade de cada classe para a determinada imagem e realizando o treinamento em vinte e cinco épocas. O intuito foi a busca de simplicidade, agilidade e eficiência na obtenção de resultados, visto que a etapa de treinamento leva tempo e, quanto mais rápido for a classificação, melhor, ainda mais para dispositivos móveis.

O método de *transfer learning* – ou transferência de aprendizado -, serve para aumentar o desempenho do modelo de CNN e, ao mesmo tempo, não ter a necessidade de obtenção de uma base de dados gigantesca, o que, em muitos casos, é um problema recorrente. Nesse método, faz-se o treinamento da rede neural a partir do *dataset* atual e, além disso, de um modelo de CNN pré-treinado e com uma grande quantidade de dados (COSTA, 2019). A técnica de *transfer learning* foi aplicada utilizando um modelo de classificação de imagem do Keras chamado MobileNetV2.

Já o método de *data augmentation* permite a geração de novas *features* a partir de manipulações na imagem que se deseja gerar novos exemplares, como, por exemplo, a mesma imagem de ponta cabeça, com zoom, com rotações etc. Esse método é muito útil para a ampliação da base de dados.

Finalmente, o *early stopping* serve para parar o treinamento assim que o modelo parar de melhorar. Isso é validado a cada época passada e a métrica verificada é a precisão. O *early stopping* é muito bom para impedir que a acurácia diminua devido ao aprendizado de ruídos das imagens (YING, 2019).

Uma função de ativação é utilizada para determinar a saída do neurônio (RODRIGUES, 2018) e como ela deve ser impressa na camada de saída. A função de ativação *softmax* é comumente utilizada nas camadas de saída para problemas que envolvem mais de duas classes, ou seja, classificações multiclasse. Essa função transforma as saídas em probabilidades que, se somadas, resultam em 1 (ARAÚJO, 2019). Desse modo, a saída que obtiver uma probabilidade maior do que as outras, é a que o algoritmo definiu para a imagem.

No início do treinamento da rede neural, são gerados, aleatoriamente, pesos que são responsáveis por determinar o nível de importância que determinada característica da camada de entrada tem em relação ao neurônio. À medida em que as épocas vão sendo passadas (no caso do presente trabalho, 25) com um *batch size* – ou não – definido (no caso, tamanho de 32), os pesos vão se reajustando e a rede vai aprendendo. Esse processo de reajuste de pesos é chamado de *backpropagation*.

A construção do algoritmo de redes neurais foi feita a partir do *Google Colab*², utilizando o *Python* como linguagem de programação.

Por fim, assim que obtidas as taxas de generalização consideradas satisfatórias, o modelo foi salvo e implementado à uma API para a comunicação com a aplicação principal.

² Ambiente de notebooks executado em nuvem. Disponível em: <https://colab.research.google.com/?utm_source=scs-index>

4.3 DESENVOLVIMENTO DO JOGO

O primeiro passo se deu pela construção da narrativa do jogo, baseada na jornada do herói, para que os objetos a serem apresentados via imagem e em tempo real possam se encaixar adequadamente ao roteiro.

Foi desenvolvido o GDD (*Game Design Document*) do jogo. O GDD é o documento que contém as informações necessárias para a criação de um jogo. Essas informações são dispostas em uma forma organizada, detalhada e de fácil entendimento, para que os outros integrantes da equipe (pensando em mais de uma pessoa) consigam absorver a ideia do produto. Esse documento contém o roteiro, mecânicas, estilo artístico e outras características importantes do jogo.

O *preview* (protótipo de visualização básicas) do jogo foi desenvolvido utilizando o motor de jogos Unity³. A linguagem dos scripts é o C#, já que ela é a usada por padrão pela Unity.

O jogo é de exploração. A proposta inicial é de que o personagem (padrão para todos os jogadores) está em uma cidade e seu objetivo é explorá-la, a fim de descobrir informações importantes e interessantes sobre as pessoas que moram lá. Para o progresso em determinadas áreas, o jogador deve apresentar itens específicos (conforme o roteiro) para a câmera do *smartphone*. Em alguns momentos do jogo, devido a exploração, é necessário realizar e validar uma ação social para conseguir informações e progredir no jogo.

Na sequência, foi iniciado o processo de criação do *game*. O tipo de arte utilizada é a *pixel art* – imagens construídas a partir de pixels visíveis e que foram colocados na tela de forma proposital (SILBER, 2016) – criada por softwares de criação de imagens vetoriais (Adobe Illustrator, Gimp, Affinity Designer, Blender, entre outros) e pixel a pixel (Adobe Photoshop, Piskel, AseSprite, Pyxel Edit, entre outros). A Unity disponibiliza configurações de imagem para a manipulação de *pixel arts*, com a remoção de filtro bilinear/trilinear e a não-compressão de imagens, o que faz com que as *pixel arts* sejam renderizadas e visualizadas em qualidade máxima, independente da escala.

³ Plataforma de desenvolvimento de jogos. Disponível em: <<https://unity.com/pt/products/unity-platform>>

O estilo artístico foi muito inspirado em jogos 16 e 32-bits, como Pokemon e Earthbound. Os *sprites* e suas animações foram desenvolvidas a partir do software Piskel, o qual é aberto e gratuito. Os elementos foram criados com uma dimensão base de 32x32 pixels e foram proporcionalmente produzidos a partir dessa dimensão. As paletas de cores, bem como a velocidade de reprodução de frames de animação foram escolhidos de forma a dar vida ao jogo e agradar a maioria dos jogadores, baseando-se em padrões encontrados em outros jogos que fizeram sucesso e que agradam uma base grande de fãs. Dentre os elementos gráficos criados para o presente projeto, estão: menus, epílogo animado, *backgrounds*, personagens, inimigos, cenários etc.

Em paralelo à parte gráfica, foram implantadas as mecânicas do *game* (definidas previamente em documentação própria, o GDD) e outros elementos *in-game*.

A partir da linguagem de programação C# e da *game engine* Unity, foi possível programar e dispor os objetos dentro da cena do jogo. Foi desenvolvido: sistema de diálogos flexível; sistema de movimentação com tratamento de estados de animação; adaptação de elementos narrativos; manipulação de *Sorting Layer*; criação de eixo customizado para trabalhar com transparência em interações *Top Down* e desenvolvimento de *outlines* a partir de *Gizmos* para definição de limites de câmera; desenvolvimento de lógica específica para a transição de cenas; persistência de informação e adaptação de *respawn*; efeitos de transição; interação entre jogador e NPC; criação de mecânicas de combate e movimentação de patrulha para inimigo; interações com teclado para validação de código (referente ao avanço da narrativa), IA de inimigo com *target* focado no jogador (atira projéteis no local em que o *player* está posicionado no momento do disparo), entre outras mecânicas.

A Figura 8 apresenta as *outlines* utilizadas para demarcar os limites de câmera a partir do ponto central do objeto que a representa.

Figura 8 - *Outlines* vermelhas a partir de *Gizmos*



Fonte: Autoria própria.

Os *Gizmos* são recursos gráficos que auxiliam o desenvolvedor a depurar a aplicação e/ou fornecem auxílio para a definição de configurações dentro da cena (UNITY, 2022).

A Figura 9 mostra o menu principal do jogo. É a primeira tela do *game*, onde o jogador pode selecionar saber mais sobre o jogo, seu funcionamento e comandos (Sobre), iniciar a aventura (Jogar) ou sair da aplicação (Sair).

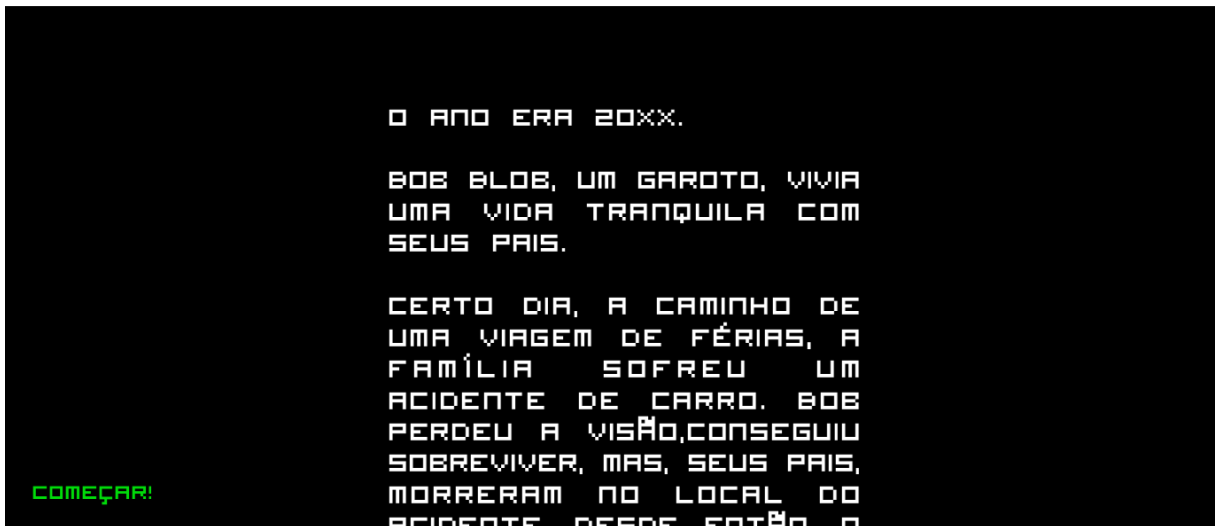
Figura 9 - Menu do jogo



Fonte: Autoria própria.

Após clicar em “Jogar”, no menu inicial, o jogador será apresentado ao universo do jogo com um pequeno prólogo da história do *game*, que terá uma movimentação constante de baixo para cima, parecido com o que se vê em filmes como Star Wars jogos como Bloodstained: Curse of the Moon, como mostrado na Figura 10.

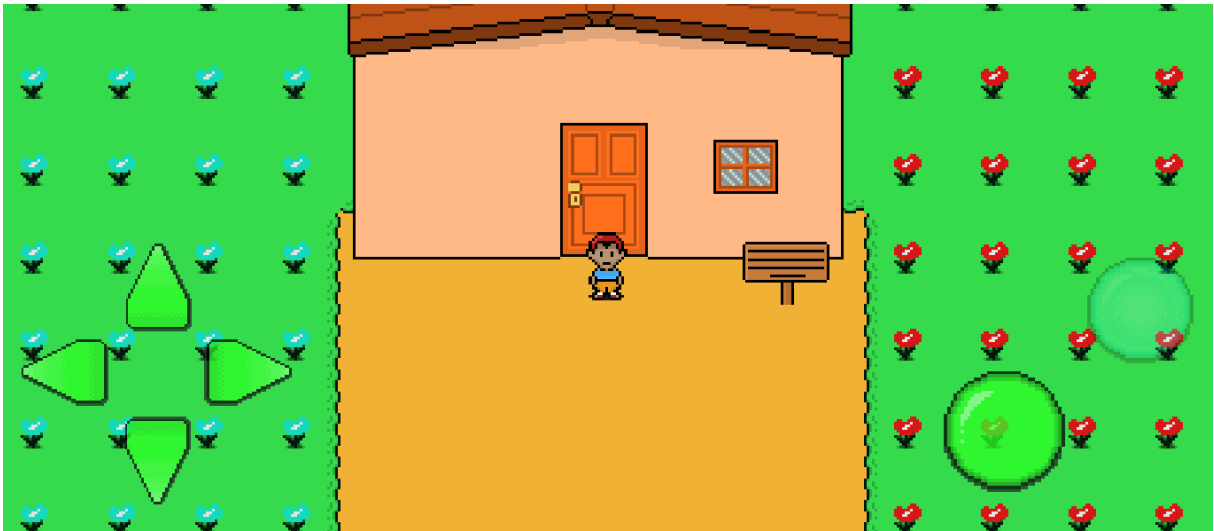
Figura 10 - Prólogo



Fonte: Autoria própria.

Ao passar pelo prólogo, esse (mostrado abaixo, na Figura 11) é o ponto inicial da aventura. O personagem será posicionado em frente ao orfanato da cidade, lugar em que ele passou parte de sua infância, até o momento da aventura. A partir desse ponto, o jogador tem controle sobre o personagem a partir do *joystick* visual projetado ao lado esquerdo e dos botões de ação e ativação da câmera ao lado direito.

Figura 11 - Momento de início do jogo.



Fonte: Autoria própria.

O sistema de diálogos foi projetado para imprimir em tela, proceduralmente, letra por letra de cada sentença. A cada frase dita pelo NPC, o jogador pode apertar o botão de ação para prosseguir para a próxima sentença. Quando o botão azul estiver com a opacidade alta (bem iluminado e forte), será possível clicar no mesmo, tirar uma foto e/ou escolher uma foto da galeria para sua utilização como camada de entrada para o algoritmo de classificação. No caso abaixo (Figura 12), o NPC pede uma caneta ao personagem, que precisa da ajuda do jogador para encontrá-la. Nesse sentido, cabe ao usuário adquirir uma foto do objeto em questão e enviar ao software, para que ela seja classificada e indique se aquele objeto apresentado é uma caneta ou não e permitir o progresso narrativo do jogador.

Figura 12 - Sistema de diálogos e classificação de imagem



Fonte: Autoria própria.

A Figura 13 mostra o segundo cenário do jogo, a floresta. Nela, o jogador será apresentado a inimigos com movimentações diferentes e a um ambiente vegetativo, que é caminho para o castelo de Lorde Rawn, vilão do jogo. Também na floresta, o usuário terá o desafio de adquirir mais uma foto de um objeto específico para a progressão na narrativa, o balde.

Figura 13 - Floresta e inimigos



Fonte: Autoria própria.

Num primeiro momento, o jogador, ao chegar ao local abaixo, terá contato com os primeiros inimigos do jogo. Os pássaros têm uma movimentação de patrulha no

eixo x e fazem parte do tutorial do sistema de combate do jogo, como representado na Figura 14.

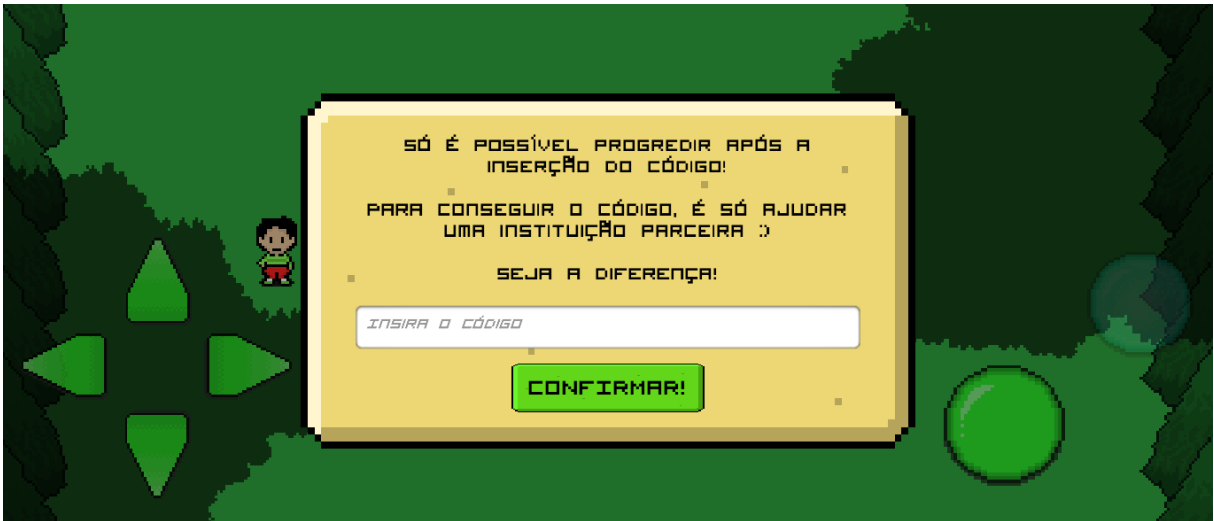
Figura 14 - Primeiros inimigos do *game*



Fonte: Autoria própria.

A Figura 15 mostra uma imagem do fim do *preview*. A modal abaixo receberá como entrada um conjunto de caracteres que, se forem iguais ao código entregue pela instituição parceira do projeto, validarão a passagem de cenário. Enquanto a modal estiver ativa, o jogador não poderá se movimentar. Informações detalhadas sobre a jogabilidade, gráficos, narrativa, entre outras coisas, estão disponíveis no **ANEXO A** deste documento.

Figura 15 - Modal para progressão de ambiente



Fonte: Autoria própria.

Outro procedimento importante foi deixar o jogo portátil para os dispositivos Android e integrar as tecnologias de redes neurais e o próprio jogo aos componentes do *smartphone* (câmera, galeria, entre outros).

Depois de testado, os métodos e técnicas utilizadas para a criação do jogo foram analisados e estudados, a fim de mostrar a importância, a funcionalidade e a viabilidade dessas técnicas para a criação e jogabilidade do produto (*preview*).

Os testes foram conduzidos em um dispositivo Galaxy J8, do ano de 2018, com processador Qualcomm Snapdragon 450 e memória RAM de 4 GB.

4.4 ARQUITETURA DAS APLICAÇÕES

A arquitetura do projeto foi pensada, principalmente, levando em consideração a experiência do usuário e a especialização de tarefas. A base da arquitetura consiste em uma API, que é responsável por carregar o modelo de redes neurais convolucionais para que seja realizada a classificação. A API comunica-se com a aplicação principal (jogo), essa, que tem a responsabilidade de processar todos os processos *in-game* (mecânicas, visual, física etc.).

A escolha de desenvolver a API em Python se deu, pois, o modelo de *machine learning* foi desenvolvido a partir da mesma linguagem, utilizando os *frameworks* Tensorflow e Keras para o pré-processamento das imagens, treinamento, validação,

e obtenção de métricas. Além disso, foi utilizada a biblioteca *matplotlib* para a plotagem de dados e gráficos, o que foi importante para o procedimento de análise.

A construção da API foi realizada a partir de um notebook do Repl.it, site que proporciona a criação e hospedagem de sites e aplicações.

Foi utilizado o *microframework* Flask para facilitar o desenvolvimento da API. O Flask proporciona ferramentas e alternativas simples para o desenvolvimento web, sem contar que proporciona o desenvolvimento de aplicações robustas.

Durante a comunicação, a API recebe um arquivo *json* que virá, originalmente, da aplicação principal. Ao receber o arquivo, a API irá desserializar o mesmo e convertê-lo para um *array* de bytes. Esse *array* de bytes será convertido para uma imagem a partir da biblioteca Pillow e, posteriormente, salvo como arquivo. A API persiste os arquivos salvos e, com isso, há a possibilidade de persistir a pasta com o modelo de *machine learning* dentro do Repl.it. Dessa maneira, não há a necessidade de utilizar um banco de dados ou serviço de armazenamento para obter as informações do modelo. Com essa solução, há a economia de recursos de processamento e de velocidade na classificação, já que não serão necessárias consultas para a obtenção de dados.

A aplicação principal, o jogo, foi construída em C# a partir da Unity. A partir de uma requisição HTTP, com a imagem importada para a aplicação, a textura 2D da imagem é convertida em bytes. Posteriormente, esse *array* de bytes é convertido em uma *string* base64 e é enviada via POST para a API, a partir de um arquivo *json*. A comunicação é eficiente e rápida entre as aplicações.

4.5 ENGAJAMENTO SOCIAL

A última parte do projeto consiste nos detalhes da implementação do engajamento social com instituições/entidades. Um código representa a chave que fica em posse de diretores das instituições de causas sociais. O jogador deve realizar ações sociais e/ou benéficas para essas instituições para conseguir acesso ao código e progredir no jogo.

A ideia foi implementada na aplicação, mas não com as entidades/instituições. No fim do *preview*, é mostrada ao usuário uma modal com um campo para inserção de texto, onde o jogador deve informar o código que, teoricamente, é dado a ele ao realizar a ação social. Essa é uma proposta que pode receber avanços em

implementações futuras e ser aplicada, também, com as instituições, mas tem suporte no projeto atual.

5. RESULTADOS OBTIDOS

Os resultados foram satisfatórios e condizentes com a proposta da pesquisa. Durante a criação do jogo, foi constatado que a Unity é uma *engine* eficiente para a construção de um jogo mobile e que permite a comunicação com APIs externas, o que foi excelente e essencial para trabalhar com a integração do algoritmo de redes neurais à aplicação.

Além disso, em relação à parte gráfica, a utilização de gráficos com dimensões base de 32x32 pixels auxiliou no desempenho da aplicação, levando em consideração que alguns aparelhos são mais antigos e que essas imagens, por serem pequenas, tem um peso menor. Outro recurso que ajudou na otimização são os *tilemaps*. Os *tilemaps* reproduzem a imagem de referência e têm muito mais desempenho em relação à construção de mapas e cenários apenas com *sprites*, o que reduz drasticamente a quantidade de *game objects* em cena.

A Figura 16 apresenta informações a respeito da diferença de desempenho entre uma cena da Unity com o uso de vários *sprites* e *tilemaps*.

Figura 16 - Diferença de desempenho entre sprites e tilemaps

Unity Scene using Sprites	Unity Scene using Tilemaps
370,000 lines in the Scene file	30,000 lines in the Scene file
10 MB Scene size	831 KB Scene size
1930 batches	157 batches
244 ms per frame	13 ms per frame
RAM usage: 1.1 GB	Ram usage: 21 MB

Fonte: Unity, 2022.

Foi analisado, também, que um algoritmo de redes neurais convolucionais é uma excelente opção para trabalhar imagens, já que o Tensorflow, por exemplo, aceita imagens na camada de entrada. A utilização do Tensorflow, também, auxiliou muito no processo de treinamento e validação, já que ele proporciona várias bibliotecas e funcionalidades que facilitam o desenvolvimento da rede e o treinamento dos modelos.

Outra biblioteca utilizada e que facilitou a análise dos resultados é a matplotlib, que tem a função de plotar imagens, gráficos, estatísticas e informações de acurácia de um jeito fácil e para uma melhor visualização.

Como citado na metodologia do presente trabalho, foram aplicados métodos como *transfer learning*, *data augmentation* na rede neural. Essas técnicas foram implementadas para evitar problemas como o *overfitting* e, além disso, deixar a rede mais robusta e preparada para as entradas.

O *overfitting* ocorre quando o modelo não consegue generalizar com eficácia os dados observados e os nunca vistos (YING, 2019).

Devido à existência do *overfitting* o modelo funciona perfeitamente no conjunto de treinamento, enquanto se ajusta mal no conjunto de teste. Isso se deve ao fato de que o modelo superajustado tem dificuldade em lidar com partes das informações no conjunto de teste, o que pode ser diferente do conjunto de treinamento (YING, 2019, p. 2, tradução nossa).

A Figura 17 apresenta a quantidade de parâmetros utilizados no treinamento. Ao total, foram treinados três mil oitocentos e quarenta e três parâmetros (*Trainable*

Params). O restante dos parâmetros (*Non-trainable params*) são referentes ao modelo usado durante o *transfer learning*.

Figura 17 - Quantidade de parâmetros usados durante o treinamento

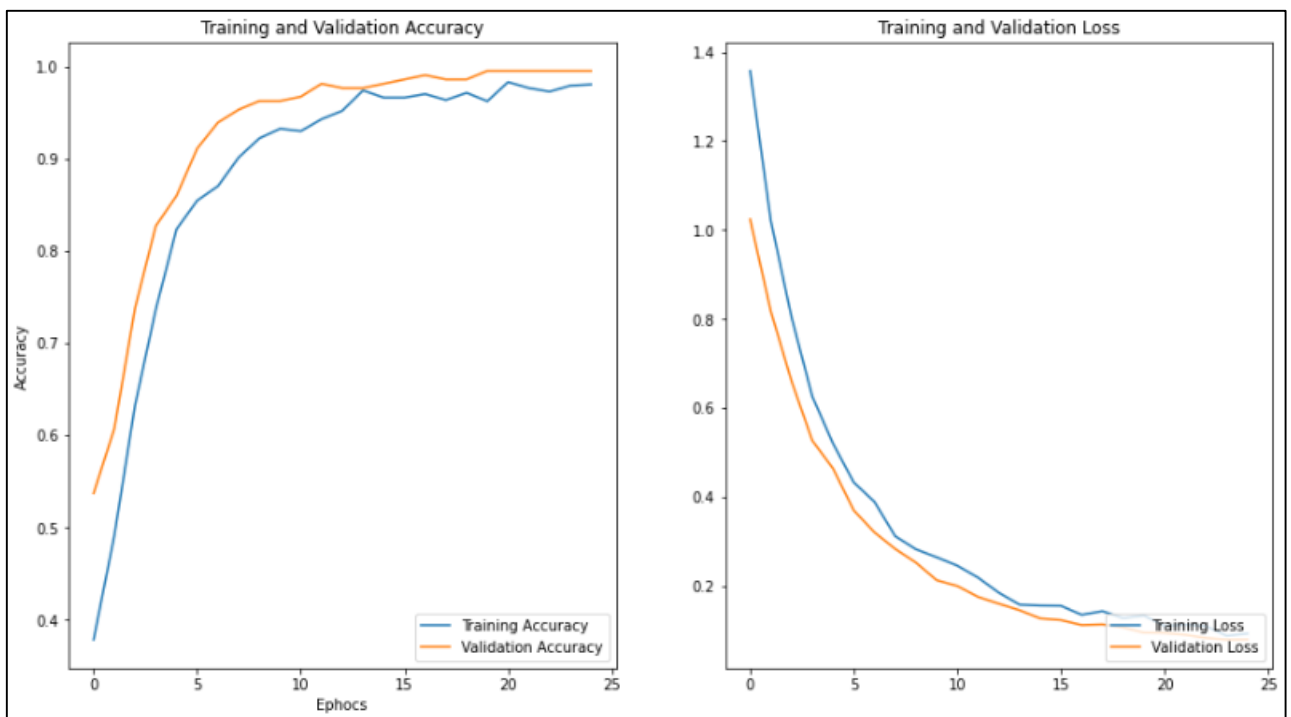
Total params: 2,261,827
Trainable params: 3,843
Non-trainable params: 2,257,984

Fonte: Autoria própria.

Foram analisadas métricas importantes e significativas para avaliar o desempenho do modelo de redes neurais convolucionais e, as mesmas, apresentaram resultados excelentes considerando o tamanho dos *datasets*.

A Figura 18 mostra que acurácia da rede foi excepcional, bem como a perda.

Figura 18 - Informações da acurácia e perda após o treinamento e validação



Fonte: Autoria própria.

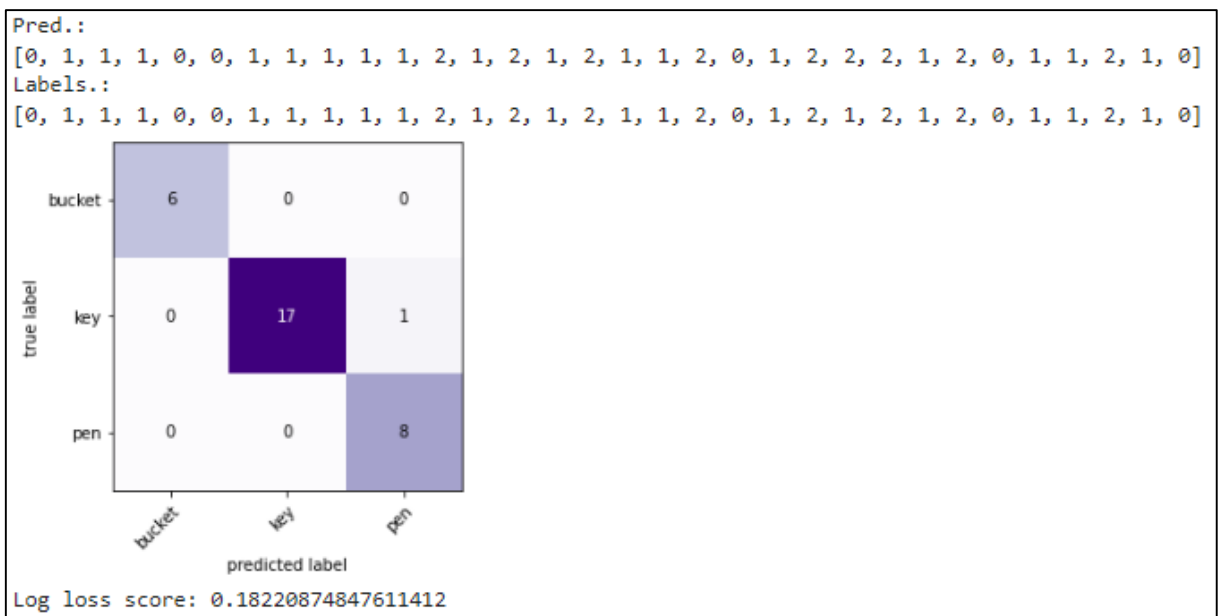
Como demonstrado nos gráficos acima, o modelo aprendeu muito bem e teve sua perda reduzida em um número pequeno de épocas (eixo horizontal). Ao fim do treinamento e validação, o modelo atingiu uma acurácia de 98% e perda de menos de 0.1%.

Além disso, as matrizes de confusão plotadas demonstraram que o modelo atingiu um número muito pequeno de predições erradas, como mostrado nas imagens abaixo:

As figuras a seguir representam as seis matrizes de confusão plotadas para avaliar o desempenho do algoritmo. O eixo Y representa as “*true labels*”, ou seja, as reais classes das imagens. Já o eixo X representa as “*predicted labels*”, que são as classes que o algoritmo previu para as imagens em questão. Além disso, as classes “*bucket*” (balde), “*key*” (chave) e “*pen*” (caneta) representam as classes dos objetos que são classificados. Além disso, para cada matriz, foi plotado um valor de perda (*log loss*).

A Figura 19 apresenta a primeira matriz de confusão plotada pelo modelo. Nela foi calculada uma perda de 0.1 devido à predição incorreta, representada no lado direito central da matriz. O algoritmo previu uma chave, quando a classe era uma caneta.

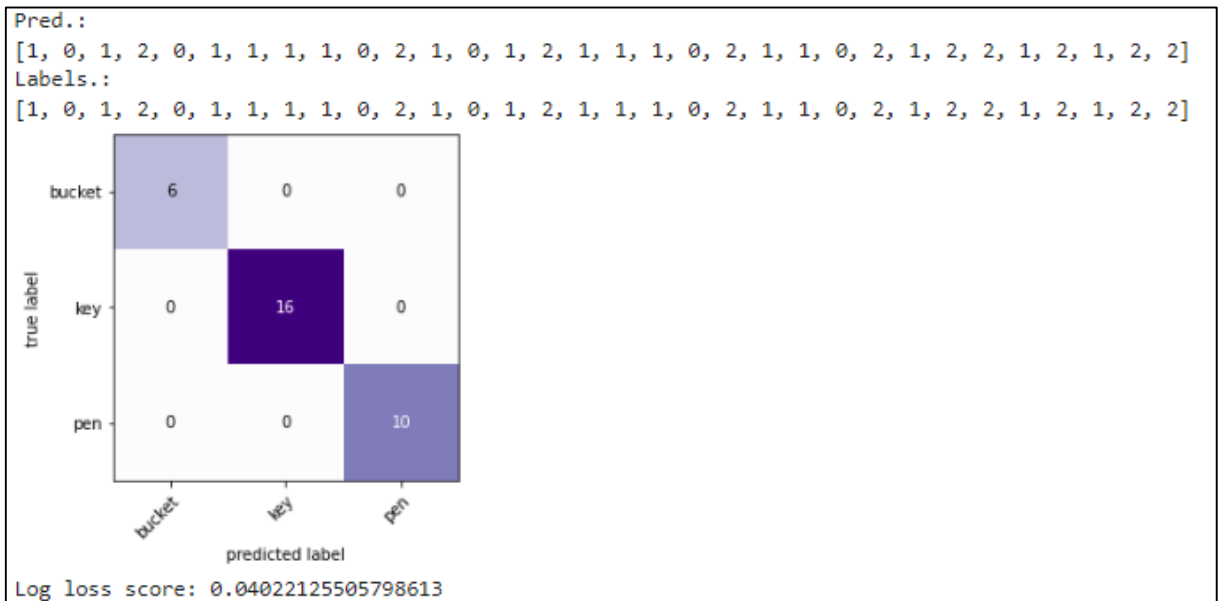
Figura 19 - Primeira matriz de confusão para validar predições do modelo



Fonte: Autoria própria.

A Figura 20 representa a segunda matriz de confusão plotada. Nela, o resultado de perda foi melhor, já que não houve predições incorretas

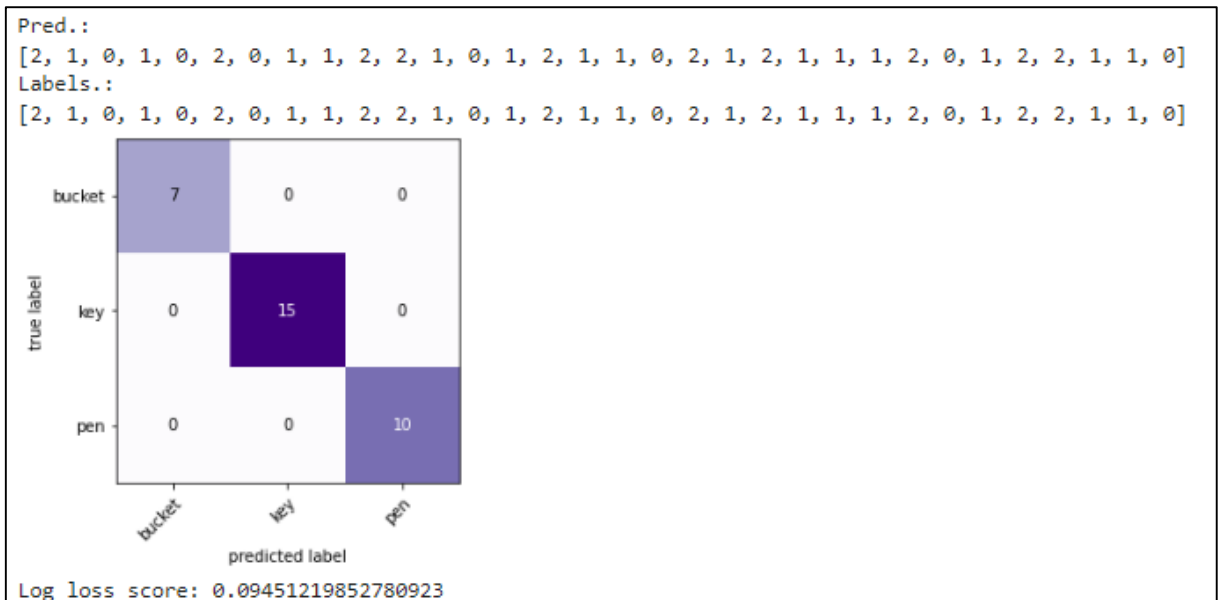
Figura 20 - Segunda matriz de confusão para validar previsões do modelo



Fonte: Autoria própria.

A Figura 21 segue sem previsões erradas. Dessa vez, foram analisadas mais imagens de balde e menos imagens de chaves do que na matriz anterior.

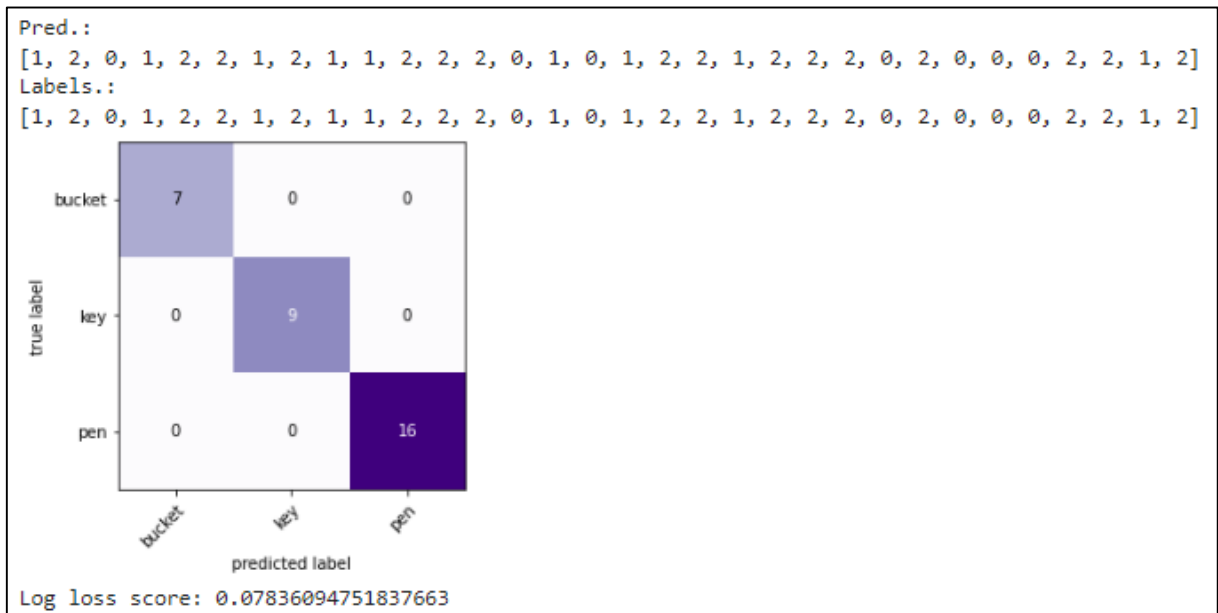
Figura 21 - Terceira matriz de confusão para validar previsões do modelo



Fonte: Autoria própria.

Já na Figura 22, a quantidade de imagens de canetas analisadas foi superior à quantidade de chaves e de baldes.

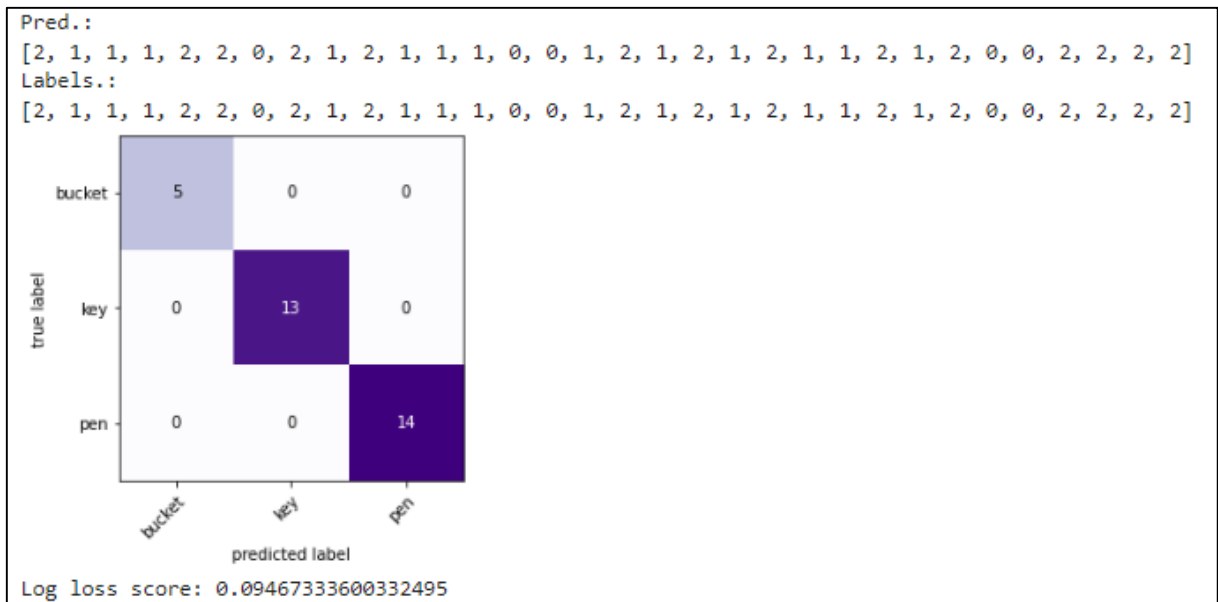
Figura 22 - Quarta matriz de confusão para validar predições do modelo



Fonte: Autoria própria.

A Figura 23 mostra a quarta matriz de confusão plotada. Nela, também, não houve predições incorretas.

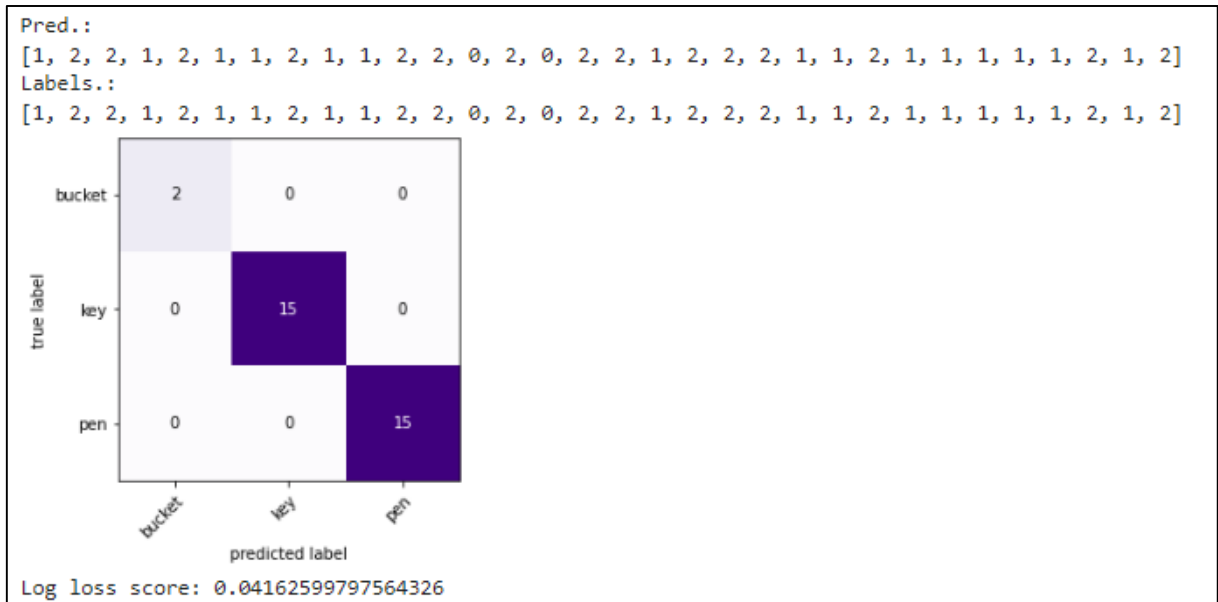
Figura 23 - Quinta matriz de confusão para validar predições do modelo



Fonte: Autoria própria.

Por fim, a sexta matriz plotada, como mostrado abaixo, na Figura 24, também segue sem erros e com uma quantia de duas imagens de baldes, quinze imagens de canetas e quinze imagens de chaves.

Figura 24 - Sexta matriz de confusão para validar predições do modelo



Fonte: Autoria própria.

Cada matriz de confusão foi criada e plotada utilizando imagens do *dataset* de validação de acordo com o *batch size* definido – que é de 32. Dessa forma, cada matriz avaliou 32 imagens aleatórias do *dataset* de validação para identificar os erros e acertos do modelo. É importante ressaltar que, em todas as matrizes, a quantia de imagens de baldes classificadas foi inferior devido ao tamanho do *dataset*, que é menor do que o de chaves e canetas. Logo, a matriz vai validar quantias de imagens das respectivas classes correspondentes proporcionais ao tamanho de cada *dataset* (até atingir os 32 exemplares).

Apenas a primeira matriz apresentou um erro de predição, o que é muito satisfatório considerando o tamanho dos *datasets*.

Também foi analisada uma média de tempo para a classificação da imagem. Foram feitos 10 testes para que a média fosse encontrada e o resultado foi de 41 segundos por classificação.

Portanto, o uso da classificação de imagens em um jogo mobile é muito possível até mesmo para *smartphones* mais antigos, considerando uma experiência do usuário satisfatória e com a classificação otimizada. Além disso, também é possível

incentivar a realização de ações sociais por meio de um jogo, já que ele foi implementado no *game* em questão.

6. DISCUSSÃO DOS RESULTADOS

De acordo com Silva *et al.* (2016), o Brasil é o 4º maior mercado mobile do mundo, com maior concentração no sistema operacional Android. Além disso, cerca de 52% das atividades realizadas nos celulares pelos brasileiros estão relacionadas a jogos (SILVA *et al.*, 2016). Desse modo, a escolha em optar pelo desenvolvimento de um jogo mobile, para a plataforma Android, mostra-se assertiva.

A escolha de um algoritmo de redes neurais convolucionais, com treinamento supervisionado, também foi satisfatória. As redes neurais permitem que, em suas camadas de entrada, sejam inseridas imagens para que - a partir de uma ou várias (Deep Learning) camadas ocultas, que vão definir pesos para parâmetros aprendidos – sejam classificadas a partir de uma ou mais camadas de saída rotuladas e definidas pelo desenvolvedor.

O framework Tensorflow se mostra muito eficaz e facilita a criação do modelo de CNN (Convolutional Neural Network). Existem muitos algoritmos de machine learning construídos através desse framework, como, por exemplo, um algoritmo de classificação de imagens de ervas (BELTRAN; FELIX; THIELO, 2017); um classificador de imagens para o aprendizado de idiomas (GUIMARÃES FILHO; LOURENÇO; DO NASCIMENTO, 2020) e um classificador de imagens dermatoscópicas para detecção de melanoma (SANTOS, 2020).

Apesar de os resultados obtidos serem satisfatórios, para projetos futuros, podem-se aplicar melhorias para aprimorar a performance do modelo de redes neurais convolucionais. Uma possibilidade para a melhora da perda e da acurácia seria o aumento do número de imagens dos *datasets* e mais variações de imagens dentre as classes (balde, chave e caneta). Essa alternativa também é mencionada no trabalho de classificação de ervas (BELTRAN; FELIX; THIELO, 2017), em que a acurácia de validação chegou a 99%. O trabalho classificador de imagens dermatoscópicas para detecção de melanoma (SANTOS, 2020) apresentou acurácia de 94% no treinamento e validação.

Dessa forma, os resultados obtidos nesse trabalho mostram que é possível a criação de jogos com tecnologias para a classificação de imagens de forma que a experiência do usuário seja satisfatória e inovadora, unindo essas tecnologias a técnicas narrativas como a quebra da quarta parede.

7. CONSIDERAÇÕES FINAIS

Foi concluído que é possível desenvolver o trabalho apresentado neste projeto de pesquisa de uma forma otimizada para inúmeros dispositivos a partir de bibliotecas *open source* e outros recursos para a criação do modelo de redes neurais convolucionais. Todo o processo de criação foi muito positivo e não houve tarefas – previstas antes do início do desenvolvimento do *preview* – que não puderam ser cumpridas.

Para implementações futuras, vale ressaltar algumas observações para melhorias e/ou novas *features*.

O aumento no número e variedade de imagens dos *datasets* resulta em um modelo de rede neural convolucional que consegue generalizar ainda mais as *features* que entram na rede, o que, possivelmente, proporcionaria uma precisão acima dos 98%.

Além disso, sugere-se o contato com instituições/entidades beneficentes para a aplicação da proposta de engajamento social, unido da elaboração de formulários avaliativos para analisar a receptividade dessa proposta pelos usuários.

8. ORÇAMENTO

Descrição	DISPONÍVEL		
	Qtd e	Valor Unit. (R\$)	Valor Total (R\$)
Notebook Acer Aspire 5, A515-52G-78HE, Intel Core i7-8565-U, 8GB de RAM, placa de vídeo da NVIDIA, GeForce MX130 com 2GB de VRAM e HD de 1TB.	1	R\$ 3.000,00	R\$ 3.000,00
Softwares opensource utilizados: Microsoft Visual Studio 2019, Unity 2019.4.17.	1	-	-
Smartphone com sistema operacional Android	1	R\$ 1.000,00	R\$ 1.000,00
TOTAL			R\$ 4.000,00

REFERÊNCIAS

ABREU, Leonardo Marques de. **Usabilidade de Telefones Celulares com base em Critérios Ergonômicos**. 2005. Disponível em: <https://www.maxwell.vrac.puc-rio.br/6705/6705_1.PDF>

ARANHA, Eduardo; ARAÚJO, Wendell; MADEIRA, Charles. **IA Aplicada a Jogos Digitais: Oportunidades e Desafios no Campo da Educação**. 2018. Disponível em: <https://sol.sbc.org.br/index.php/desafie/article/view/3105>. Acesso em 13 ago. 2022.

ARAÚJO, Aline Moura. **CLASSIFICAÇÃO E DETECÇÃO DE PESSOAS EM AMBIENTES NÃO CONTROLADOS UTILIZANDO REDES NEURAI CONVOLUCIONAIS**. 2019. Disponível em: <<https://repositorio.ufpb.br/jspui/bitstream/123456789/16098/1/AMA02102019.pdf>>. Acesso em: 13 ago. 2022.

BARRETO, Jorge M.. **Introdução às Redes Neurais Artificiais**. 2002. Disponível em: <http://www.inf.ufsc.br/~j.barreto/tutoriais/Survey.pdf>

BATISTA, Thiago José de Souza et al. **Um estudo sobre a história dos jogos eletrônicos**. 2007. Disponível em: <http://re.granbery.edu.br/artigos/MjQ4.pdf>

BELTRAN, Rafael; FELIX, Eduardo Bianchin; THIELO, Marcelo Resende. TENSOR FLOW FOR POETS: CLASSIFICAÇÃO DE ERVAS CULTIVADAS EM CASA. **Anais do Salão Internacional de Ensino, Pesquisa e Extensão**, v. 9, n. 2, 2017.

BLACK LIVES MATTER: ENTENDA MOVIMENTO POR TRÁS DA HASHTAG QUE MOBILIZA ATOS, UOL. 2020. Disponível em: <https://www.uol.com.br/universa/noticias/redacao/2020/06/03/black-lives-matter-conheca-o-movimento-fundado-por-tres-mulheres.htm>. Acessado em 19 mar. 2021

BONSIEPE, Gui. **Design: como prática de projeto**. São Paulo: Editora Edgard Blucher, 2012.

BRASIL TEM 424 MILHÕES DE DISPOSITIVOS DIGITAIS EM USO REVELA A 31ª PESQUISA ANUAL DO FGVCIA, FGV, 2020. Disponível em: <https://portal.fgv.br/noticias/brasil-tem-424-milhoes-dispositivos-digita-31a-pesquisa-anual-fgvcia>. Acessado em 19 mar. 2021

CAMPBELL, Joseph. **O herói de mil faces**. 10. ed. São Paulo: Pensamento, 1997.

CARVALHO, Gabriel Rios De. **A importância dos jogos digitais na educação.**

2018. Disponível em:

<https://app.uff.br/riuff/bitstream/1/8945/1/TCC_GABRIEL_RIOS_DE_CARVALHO%20%281%29.pdf>

CERRI, Ricardo; CARVALHO, André Carlos Ponce de Leon Ferreira de. **Aprendizado de máquina: breve introdução e aplicações.** 2017. Disponível em:

<https://ainfo.cnptia.embrapa.br/digital/bitstream/item/184785/1/Aprendizado-de-maquina-breve-introducao.pdf>

COSTA, Suayder Milhomem. **USO DE TRANSFERENCIA DE APRENDIZADO E REDE NEURAL SEM PESO PARA DETECÇÃO DE IMAGENS DE DEFEITOS EM VIAS PAVIMENTADAS.** 2019. Disponível em: <

<https://repositorio.uft.edu.br/bitstream/11612/3198/1/Suayder%20Milhomem%20Costa%20-%20TCC.pdf>>. Acesso em 13 ago. 2022.

GIZMOS, UNITY. 2022. Disponível em:

<<https://docs.unity3d.com/ScriptReference/Gizmos.html>>. Acessado em 8 mar. 2022.

GUIMARÃES FILHO, Marcos G.; LOURENÇO, Rafael P.; DO NASCIMENTO, Anderson S. Prisma: Um classificador de imagens para aprendizado de idiomas. In: **Anais da VI Escola Regional de Sistemas de Informação do Rio de Janeiro.** SBC, 2020.

HAYKIN, Simon. **Redes Neurais: Princípios e práticas.** 2007. Disponível em:

[https://books.google.com.br/books?hl=pt-](https://books.google.com.br/books?hl=pt-BR&lr=&id=bhMwDwAAQBAJ&oi=fnd&pg=PP1&dq=redes+neurais+&ots=08rrlHLSFp&sig=epo03llu1dATD3ajDEQWgqR90cc#v=onepage&q=redes%20neurais&f=false)

[BR&lr=&id=bhMwDwAAQBAJ&oi=fnd&pg=PP1&dq=redes+neurais+&ots=08rrlHLSFp&sig=epo03llu1dATD3ajDEQWgqR90cc#v=onepage&q=redes%20neurais&f=false](https://books.google.com.br/books?hl=pt-BR&lr=&id=bhMwDwAAQBAJ&oi=fnd&pg=PP1&dq=redes+neurais+&ots=08rrlHLSFp&sig=epo03llu1dATD3ajDEQWgqR90cc#v=onepage&q=redes%20neurais&f=false)

HIGUCHI, Adriane Aparecida da Silva. **Tecnologias móveis na educação.** 2011.

Disponível

em:

<<http://tede.mackenzie.br/jspui/bitstream/tede/1818/1/Adriane%20da%20Silva%20Higuchi.pdf>>

ICE BUCKET CHALLENGE DRAMATICALLY ACCELERATED THE FIGHT AGAINST ALS, ALS ASSOCIATION. 2019. Disponível em:

<https://www.als.org/stories-news/ice-bucket-challenge-dramatically-accelerated-fight-against-als>. Acessado em 19 mar. 2021

LUCESSE, Fabiano; RIBEIRO, Bruno. **Conceituação de Jogos Digitais**. 2009. Disponível em: <https://www.dca.fee.unicamp.br/~martino/disciplinas/ia369/trabalhos/t1g3.pdf>

MARTINEZ, Monica. **Jornada do Herói: estrutura narrativa mítica na construção de histórias de vida em jornalismo**. São Paulo: Annablume, 2008.

MATOS, Leni *et al.* **Evolução da tecnologia de telefonia móvel e estudo e caracterização de um sistema móvel 5G de quinta geração**. 2019. Disponível em: <https://webcache.googleusercontent.com/search?q=cache:jX7la50xFPsJ:https://periodicos.uff.br/engevista/article/view/27028/16398+&cd=3&hl=pt-BR&ct=clnk&gl=br>

MITCHELL, Tom M. **The Discipline of Machine Learning**. 2006. Disponível em: <http://ra.adm.cs.cmu.edu/anon/usr0/ftp/anon/ml/CMU-ML-06-108.pdf>

MOHRI, Mehryar; ROSTAMIZADEH, Afshin; TALWALKAR, Ameet. **Foundations of machine learning**. 2. ed. MIT press, 2018.

NAKABAYASHI, Luciana Akemi. **A contribuição da Inteligência Artificial (IA) na Filosofia da Mente**. 2009. Disponível em: <https://tede2.pucsp.br/bitstream/handle/18236/1/Luciana%20Akemi%20Nakabayashi.pdf>

NAKAMURA, Ricardo. **Design de jogos e a experiência de exploração de espaços**. 2013. Disponível em: <https://webcache.googleusercontent.com/search?q=cache:VPot0tGvfQMJ:https://dialnet.unirioja.es/descarga/articulo/4945224.pdf+&cd=4&hl=pt-BR&ct=clnk&gl=br>

OLIVEIRA, Bruno Alberto Soares *et al.* **Sistema para Classificação de Grãos de Café: Uma Comparação entre Redes Neurais Convolucionais e Processamento Digital de Imagens com MLP**. 2019. Disponível em: <https://proceedings.science/sbai-2019/papers/sistema-para-classificacao-de-graos-de-cafe--uma-comparacao-entre-redes-neurais-convolucionais-e-processamento-digital-d> >. Acesso em 13 ago. 2022.

PEREIRA, Rafaella Almeida; MOTA, Rosilane Ribeiro de. **Análise do jogo Pokemon GO e a relação dos jogadores com o espaço urbano**. 2017. Disponível em: <https://ainfo.cnptia.embrapa.br/digital/bitstream/item/184785/1/Aprendizado-de-maquina-breve-introducao.pdf>

PIRES, Joel. **Classificação de Imagens de Sementes com Deep Learning implementada na plataforma Android.** 2020. Disponível em: <https://sol.sbc.org.br/index.php/erbase/article/view/15470>. Acesso em 13 ago. 2022.

POKEMON GO ARRECADOU MAIS DE US\$ 3,6 BILHÕES EM QUATRO ANOS, TECMUNDO. 2020. Disponível em: < [https://www.tecmundo.com.br/cultura-geek/154943-pokemon-go-arrecadou-us-3-6-bilhoes-quatro-anos.htm#:~:text=Planos%20de%20Internet-,Pok%C3%A9mon%20GO%20arrecadou%20mais%20de%20US%24%203,6%20bilh%C3%B5es%20em%20quatro%20anos&text=De%20acordo%20com%20um%20relat%C3%B3rio,\(US%24%20905%20milh%C3%B5es\).>](https://www.tecmundo.com.br/cultura-geek/154943-pokemon-go-arrecadou-us-3-6-bilhoes-quatro-anos.htm#:~:text=Planos%20de%20Internet-,Pok%C3%A9mon%20GO%20arrecadou%20mais%20de%20US%24%203,6%20bilh%C3%B5es%20em%20quatro%20anos&text=De%20acordo%20com%20um%20relat%C3%B3rio,(US%24%20905%20milh%C3%B5es).>). Acessado em 29 mar. 2021.

RODRIGUES, Diego Alves. **Deep Learning e Redes Neurais Convolucionais: Reconhecimento automático de caracteres em placas de licenciamento automotivo.** 2018. Disponível em: < <https://repositorio.ufpb.br/jspui/bitstream/123456789/15606/1/DAR20052019.pdf>>. Acessado em 13 ago. 2022

SANTOS, Julio Cezar Alves dos *et al.* **Classificação de imagens dermatoscópicas com machine learning para a detecção de melanoma.** 2020.

SILBER, Daniel. **Pixel art for game developers.** Boca Raton: CRC Press, 2016.

SILVA, Leonardo Ferreira *et al.* O crescimento dos jogos no mercado mobile e suas acessibilidades. **Revista Caribena de Ciencias Sociales,(2016 02),** 2016.

SILVA, Maycon Prado Rocha *et al.* **Jogos Digitais: definições, classificações e avaliação.** 2009.

Disponível em: <http://files.projeto-reinventando-e-e-p-j-m.webnode.com/200000017-27f3c28f2c/jogos.pdf>

SILVA, Rodrigo Emerson Valentim da. **Um estudo comparativo entre redes neurais convolucionais para a classificação de imagens.** 2018. Disponível em: < https://repositorio.ufc.br/bitstream/riufc/39475/1/2018_tcc_revsilva.pdf>. Acessado em 13 ago. 2022.

SOUZA, F.; SOARES FLORES, E. PROCESSAMENTO E ANÁLISE DE IMAGENS PARA CLASSIFICAÇÃO AUTOMÁTICA DE OBRAS ARTÍSTICAS. **Anais do Salão Internacional de Ensino, Pesquisa e Extensão,** v. 11, n. 2, 28 ago. 2020.

STEPHANOU, Luis; MÜLLER, Lúcia Helena; CARVALHO, Isabel Cristina de Moura. **Guia para elaboração de projetos sociais**. 2003. Disponível em: <https://aplicacoes.mds.gov.br/sagirms/ferramentas/docs/guia-para-elaboracao-de-projetos-sociais.pdf>

TAROUCO, Liane Margarida Rockenbach, *et al.* **Jogos Educacionais**. 2004. Disponível em: <https://www.lume.ufrgs.br/bitstream/handle/10183/12990/000572691.pdf>.

VOGLER, Christopher. **A jornada do escritor: estruturas míticas para escritores**. 2. ed. Rio de Janeiro: Nova Fronteira, 2006.

YING, Xue. **Na Overview of Overfitting and its Solutions**. 2019. Disponível em: <https://iopscience.iop.org/article/10.1088/1742-6596/1168/2/022022/pdf> >. Acesso em: 13 ago. 2022.

ANEXO A – GAME DESIGN DOCUMENT – SOMEBODY

Título da demo: Somebody.

Plataforma: Celular (Android).

Gênero: Ação e aventura; exploração.

Público-alvo: Pessoas de todas as idades que tenham interesse em trabalho social e auxílio a quem necessita; pessoas de 13-25 anos, principalmente.

Sinopse: O ano era 20XX. Bob Blob, um garoto, vivia uma vida tranquila com seus pais. Certo dia, a caminho de uma viagem de férias, a família sofreu um acidente de carro. Bob perdeu a visão, conseguiu sobreviver, mas, seus pais, morreram no local do acidente. Desde então, o menino foi acolhido por seus avós, que, após alguns meses, já não tinham mais como cuidar dele, pois estavam muito velhos. O garoto foi para um abrigo e ficou lá durante anos.

Um dia, a cidade toda foi exposta a um feixe de luz muito forte e, após esse dia, todos começaram a agir de maneira muito estranha. Bob, por não ser afetado (devido ao acidente) começou a questionar as pessoas do abrigo. As cuidadoras o pediram para ficar quieto e não perguntar mais nada, apenas seguir os comandos do Lorde Rawn, que era o líder de todos. Ainda, o tiraram de seu bastão guia, que era o instrumento que o ajudava a andar por aí. Sem isso, era muito difícil ele conseguir andar muito longe sem se perder. De repente, Bob sentiu a presença de um guardião, que prometeu guiá-lo em uma jornada, de muitos desafios, para descobrir o que foi feito com o povo da cidade de Malman e impedir que coisas ruins sejam realizadas.

Cenários (demo):

- **Cidade:** Local onde o personagem principal mora. Chama-se Cidade de Malman, com 34 mil habitantes e a maioria dos serviços básicos das cidades. Local rodeado com bastante vegetação e uma floresta ao lado.

- **Floresta Vulture:** Localizada ao norte de Malman. A Floresta Vulture influencia na economia da cidade e tem uma fauna e flora impecáveis. A floresta é um tanto quanto escura, já que as árvores bloqueiam grande parte da luz do sol, mas, ainda sim, muito visível. Sua vegetação de cor verde escura é predominante e animais como macacos, aves e até panteras circulam por lá.
- **Casas:** As casas de Malman são, em sua grande maioria, de concreto, porém, existem casas de madeira, também. A mobília é de madeira e os itens são diversos.
- **Mercado:** O mercado, em Malman, é bem grande e tem vários corredores. Lá, é possível encontrar os mais variados produtos das mais variadas marcas. A área do mercado era em forma retangular, com altura maior que largura. As prateleiras acompanhavam a altura e tinham “gaps” entre elas para a passagem dos clientes. Na extremidade de cima, há o açougue, a padaria e o hortifruti.

Personagens (demo):

- **Bob Blob:** Um garoto de 13 anos que, após sofrer um acidente de carro, perdeu a visão e seus pais. Ele, então, foi morar com seus avós, mas os mesmos não tinham mais condições de cuidar dele, pois estavam muito velhos. Após isso, ele foi para um abrigo, onde pode receber todo o auxílio necessário. Bob tem estatura média, é negro, tem cabelos negros, olhos castanhos e seu maior sonho é poder enxergar seus pais novamente.
- **Tiny Robert:** Um garoto de 15 anos que se torna amigo de Bob durante a jornada do mesmo. Ele é deficiente visual, assim como Bob e também não foi afetado pelo feixe de luz que transformou a cidade de Malman. Junto a outras pessoas, Robert cria uma resistência contra o comportamento de todos na cidade, regrados pelo Lorde Rawn. Robert tem estatura baixa, é branco, tem cabelos negros, olhos castanhos e quer acabar com os planos de Lorde Rawn, o qual tem um conflito secreto.

- **Lorde Rawn:** Homem de 46 anos. Rawn é ganancioso e narcisista. Seu maior propósito é ser reconhecido. Ele gosta de idolatria e sempre foi uma pessoa egoísta e individualista. Não gosta que o toquem, nem para um abraço. Perdeu sua família enquanto era bebê e não conhece as coisas boas da vida. Foi criado por pessoas más e sempre sofreu em sua infância. Rawn tem estatura alta, é branco, cabelos castanhos, olhos negros, é narcisista, egocêntrico, egoísta e odeia contato físico.
- **Guardião:** A representação do jogador. É o elemento narrativo imersivo responsável por “levar” o jogador ao mundo fictício do *game*. O guardião tem a responsabilidade de proteger Bob e ajudá-lo a impedir que os planos do Lorde Rawn sejam realizados.

Narrativa (demo): Durante o ano de 20XX, Bob Blob, um garoto de 8 anos, vivia sua vida com seus pais. Eles eram felizes e sempre foram unidos. Durante o mês de julho, período de férias escolares, os pais de Bob conseguiram uns dias de folga para viajar com o filho. Infelizmente, no meio do caminho, aconteceu um acidente de carro e os pais de Bob morreram. O menino sobreviveu, porém, perdeu a visão. Seus avós maternos, então, cuidaram do menino por 2 anos, porém, devido à idade e à quantia de cuidados necessários que Bob precisava, os avós tiveram que deixar o neto em um abrigo.

Bob foi muito bem acolhido pelos donos do abrigo e aprendeu a viver lá. Após 3 anos vivendo no abrigo, algo diferente aconteceu. Durante uma manhã, um feixe de luz, extremamente forte, foi a vista de todos os moradores da cidade de Malman. Mesmo aqueles que estavam dormindo ou somente com o olho fechado foram assolados pelo feixe. Porém, devido à sua condição visual, Bob não foi afetado pela luminosidade. Após aquele momento, que durou milissegundos, Bob percebeu que os moradores do abrigo estavam agindo de forma diferente e quis perguntar às cuidadoras o que tinha acontecido. Uma das cuidadoras foi muito rude com o garoto e disse para ele não falar mais sobre isso e apenas seguir as ordens de um tal de Lorde Rawn. Após dizer isso, ela imediatamente confiscou seu bastão guia, o qual usava para se localizar. O menino ficou em estado de pânico e não sabia o que fazer. Ele estava se sentindo frágil e sem ânimo para seguir em frente. De repente, uma esfera azul e cintilante surgiu em sua frente. Tratava-se de um guardião, o qual surgiu para alertar Bob sobre os desafios que ele deveria enfrentar. O guardião responde a

Bob quem ele é, porque estava lá e o encoraja a seguir em frente. Depois desse diálogo, Bob sai do abrigo e começa a explorar a cidade, em busca de informações. Eram muitas dúvidas: quem é Lorde Rawn? O que era aquele feixe de luz? Foi ele que causou esse comportamento nas pessoas? O que ele pretende com isso? Por que Bob não foi afetado? Tudo isso, será respondido com o tempo.

A cidade de Malman é pequena. Uma cidade com 34 mil habitantes. Por todos os lugares que Bob andava, as pessoas falavam de uma maneira estranha e sem muito sentido.

Bob vai até a loja da cidade. Porém, antes de entrar na loja, Bob é atacado por alguns pássaros. Parece que o comportamento afetou os animais, também! Os pássaros estavam violentos e querendo atacar Bob. Nessa parte, Bob aprende o básico do que fazer quando entra em uma situação de perigo: correr! O garoto é inocente e não tem violência no coração, muito menos para ferir animais! Durante os ataques, Bob, com a ajuda do seu guardião, aprende a se esquivar dos animais que encontra. Ao se esquivar dos animais, Bob entra na loja e questiona o atendente que está lá. As pessoas estavam comprando coisas para levar para o Lorde, alegando que tudo deveria ser dele, já que o fim estava próximo! O garoto começou a suar frio, mas o guardião o acalmou. Há muitas dúvidas que o guardião ainda não sabia responder.

Saindo da loja, a próxima parada era voltar para o orfanato, agora, com o guardião, para colher pistas. Porém, ele estava fechado. Bob achou muito estranho. Como o orfanato poderia estar fechado em plena manhã? De qualquer forma, o garoto e seu guardião vasculham a cidade toda em busca de pistas. Quando entram na casa de um senhor, que é surdo, ele vê o menino e já diz que não consegue ouvi-lo. Pede para que Bob, por favor, escreva o que quer em um livro, aberto em cima de uma mesa da casa dele, para o mesmo ler. Bob, no entanto, não tinha nenhuma caneta com ele. O guardião, então, teve que encontrar uma caneta e entregá-la a Bob. Feito isso, Bob pergunta ao senhor qual era a relação do Lorde Rawn com tudo isso e por que o povo de Malman estava daquele jeito. O senhor responde que o feixe de luz emitido por Lorde Rawn é o começo do novo mundo e que todos iriam deixar esse mundo para o Lorde conquistar. Além disso, ele comenta sobre uma resistência e que o povo da cidade estava procurando-a. Era claro que o senhor tinha sido afetado também. Bob e o guardião ainda não tinham pistas o suficiente para saber o que estava acontecendo e resolvem ir até a resistência, mas, antes, tentam entrar em uma

casa em busca de informações. A porta estava trancada. O guardião, então, o ajuda, encontrando uma chave. Bob abre a porta e encontra um pedaço de jornal no chão, com a ajuda do guardião, sempre o guiando. Ele, imediatamente, lembra de seus pais e lamenta o ocorrido. Era difícil pra ele saber que nunca mais iria poder abraçar seus pais novamente. De qualquer forma, os dois descobrem a localização da resistência e resolvem ir até lá. Ao entrar no local, Tiny Robert, explica a Bob que ele fundou uma resistência e que não tinha sido afetado pelo feixe luminoso (porém, ele era o único membro). Robert conta que quer acabar com os planos de Lorde Rawn por conflitos pessoais e não conta quais conflitos são esses. Um pouco resistente, Robert deixa Bob ir para a floresta com ele. Por que a floresta? Porque lá é o caminho para o castelo de Rawn. Bob ainda fica confuso de como Robert sabe tanto em tão pouco tempo sobre esse tal Lorde, mas resolve segui-lo, já que três cabeças pensam melhor do que duas. Ao sair, o guardião deve realizar uma contribuição social, para auxiliar o grupo (e, conseqüentemente, as pessoas que precisam). Ao saírem da floresta, a demo acaba.

Desafios (demo/jogo completo):

- **Desafio Principal:** O desafio geral do jogo é impedir que o Lorde Rawn alcance seus objetivos e salvar o povo de suas garras.
- **Desafios Secundários:** O jogo contém desafios principais a serem realizados. Esses desafios envolvem o progresso da trama. Dentre os desafios, estão: passar por inimigos e chefes, conversar com NPCs para a progressão, apresentar itens aos NPCs (machine learning) e realizar trabalho social.
- **Passar por inimigos e chefes:** Nesse jogo, o jogador não luta. Ele apenas passa por inimigos e chefes e desvia deles para, em muitas vezes, tontear/cansá-los. Todos os inimigos desse jogo são enfrentados assim, mas com mecânicas e estratégias diferentes.
- **Conversar com NPCs:** Será necessária a interação com NPCs para a coleta de informações e progressão no jogo.
- **Apresentar itens aos NPCs:** Por meio de IA, com Machine Learning, o jogador deve apresentar itens que correspondam ao mesmo item, virtualmente, para permitir o progresso do jogador, considerando a trama envolvida.

- **Realizar trabalho social:** O jogo pode ser dividido em cenários. Para a transição de cada cenário (cidade, floresta, oceano...), era necessária a realização de uma ação social pelo jogador. Essa ação vai ser validada por um código e vai permitir o progresso na narrativa.

Itens/Recompensas (demo): Será possível mostrar itens à câmera do celular e auxiliar o personagem principal no progresso do jogo. Esses itens serão reconhecidos via Machine Learning. São Eles:

Nome	Cor do fundo	Interação
Caneta	Branco	Cidade de Malman (Senhor com deficiência auditiva)
Chave	Branco	Cidade de Malman (Porta trancada)
Balde	Branco	Floresta (Poço)

Vitória do jogo (jogo completo): A vitória do jogo será realizada quando o jogador interromper os planos do Lorde Rawn, salvar sua alma corrompida, salvar os

habitantes de Malman e o desfecho de cada personagem da resistência for feito. Para isso, o jogador deve enfrentar a última batalha e sair do Castelo que começa a desmoronar.

Estilo de arte: A arte utilizada será a pixel art. A escolha do estilo se dá por questão de gosto pessoal e por representar uma temática retrô, combinando os elementos sonoros e gráficos. Jogos famosos como Super Mario Bros 3, Castlevania: Symphony of The Night, Super Metroid e outros utilizam esse estilo artístico.

Sonorização: O jogo contará com trilha sonora própria, produzida a partir de Digital Audio Workplace, com características 16/32 bits. Cada trilha e efeito sonoro será elaborado com base em ambientação específica. Um cenário de oceano terá uma trilha com xilofones, teclados e ritmos mais calmos, por exemplo.

Mecânicas e jogabilidade:

- **Movimentação do personagem:** O personagem será movimentado pelo jogador através de um joystick (asset da própria Unity) de touch. A forma visual de um joystick será mostrada ao jogador, no canto inferior esquerdo, com opacidade reduzida.
- **Botão de ação (mundo virtual):** Botão responsável por inspecionar elementos e falar com NPCs. É o botão de ação do personagem no mundo virtual.
- **Botão de ação (mundo real):** Botão que aciona a câmera do dispositivo, para o jogador poder mostrar objetos na tela (que serão reconhecidos via Machine Learning e Redes Neurais) e QR Codes. O botão aparecerá em momentos chave, em que a ação de mostrar o objeto correto à câmera do celular será essencial para o progresso.
- **Câmera do jogo:** A câmera do jogo acompanhará o personagem principal (nos eixos x, y e z, proporcionalmente). A posição da câmera pegará o pivot do personagem principal (posicionado no centro do *Game Object*) como referência.

- **Sistema de diálogos:** O sistema de diálogos será “chamado” em cada interação com NPCs e objetos. O sistema pode ser dividido em duas partes: a parte gráfica e a engenharia por trás.
 - Gráfica: A caixa de diálogo será feita em formato de retângulo, na horizontal, que aparecerá com uma animação quando a interação for feita. A partir disso, as letras vão aparecendo, uma por vez, até que as frases sejam completadas. Quando uma frase for completada, uma flechinha será instanciada, indicando que o jogador pode apertar o botão de interação novamente, para avançar o diálogo.
 - Engenharia: As frases serão pré-escritas a partir de um Scriptable Object, ou seja, uma classe será criada, em C#, que armazenará os vetores de string (frases) de cada diálogo. Em outro script, será feito uma coroutine que transformará cada vetor de string em um vetor de char, e isso vai separar as letras de cada palavra, para serem instanciadas a partir de uma quantia de tempo, definida pelo programador, dentro de um foreach loop. Também serão criados métodos para o início da coroutine (trigger do diálogo) e término do diálogo, com o fechamento da caixa de diálogo e o “reset” do texto que foi armazenado com o diálogo que tinha acabado de ser mostrado. A ideia desse sistema é otimizar os diálogos e deixá-los versáteis e simples de reproduzir com outros *Game Objects*.

- **Menu do jogo:** O menu principal da demo terá o título do jogo e três principais opções: o “Início”, o “Continuar” e o “Sobre”. O primeiro inicia uma nova aventura. O “Continuar”, continua o progresso salvo e a última opção mostra um pouco sobre os desenvolvedores e como o jogo foi criado.

- **Ações sociais:** As ações sociais serão requisitadas ao jogador em um momentos da trama, nessa demo: ao sair da floresta. Para a progressão, é necessário a realização da ação. Para validar a atividade, as instituições parceiras receberão um código universal. Esse código deverá ser apresentado à câmera do celular e isso validará a passagem do mesmo ao próximo cenário. Ao chegar ao final do cenário, o jogador receberá um aviso de que só é possível progredir se uma ação social for realizada. Inserindo o código correto, o jogador será levado à tela final da demo.

Regras (demo/jogo completo): O jogo conta com algumas regras importantes:

- Não é possível passar por cenários específicos sem realizar trabalho social;
- Não é possível progredir na trama sem a apresentação de itens com IA;
- O progresso só será salvo se o jogador salvar com o guardião;
- O jogador NÃO luta com os inimigos;
- O jogador NÃO pode se movimentar ou utilizar algum comando durante o diálogo com NPCs;
- Por mais que a área seja grande, o jogador NÃO pode ultrapassar os limites dos cenários;
- Para terminar a demo, o jogador DEVE realizar a última ação social.