

Framework Procedural para Geração de Dados Sintéticos Baseados em Simulação Estatística

Uma Abordagem para Análise de Comportamento Humano

Jáder Louis de S. Gonçalves
Universidade Federal de Rondônia
Departamento de Ciência da Computação
Porto Velho, RO, Brasil
jaderlouis@proton.me

Nicolas F. C. Sales
Universidade Federal de Rondônia
Departamento de Ciência da Computação
Porto Velho, RO, Brasil
nicolascavalcante0101@gmail.com

Wyllgner F. Amorim
Universidade Federal de Rondônia
Departamento de Ciência da Computação
Porto Velho, RO, Brasil
wyllgner_franca@hotmail.com

Lucas Marques da Cunha
Universidade Federal de Rondônia
Departamento de Ciência da Computação
Porto Velho, RO, Brasil
lucas.marques@unir.br

ABSTRACT

Data sets for training convolutional neural networks (CNNs) pose a challenge due to the difficulty of obtaining them and the sensitive nature of the information involved, leading to high costs, scarcity, and potential biases. To address these limitations, synthetic data generation provides a viable solution, enabling controlled and ethical research in human behavior. The main objective was to develop a framework capable of simulating environments and generating data procedurally for AI training, focusing on engagement monitoring and the identification of behavioral deviations in educational and corporate contexts. Among the specific objectives, the creation of an application persona, *Visage Track*, stands out, designed to calculate engagement and behavioral deviations, thereby validating the developed techniques. The proposed approach integrates procedural generation methods into a complex statistical simulation, allowing for customizable and controlled scenarios without the need for direct interaction with real subjects. The framework and *Visage Track* were constructed to facilitate experiments in simulated environments. In the evaluation, the prototype demonstrated high accuracy in presence detection (up to 98%), consistent behavioral simulations across multiple scenarios, and the ability to generate scalable, multimodal datasets. These results indicate the framework's potential to support research in multimedia content generation, multimodal interaction, and affective computing, offering an adaptable and ethical infrastructure for studies involving human behavior in controlled environments.

1 INTRODUÇÃO

O treinamento de redes neurais convolucionais (*Convolutional Neural Networks* - CNNs) e outros modelos de inteligência artificial (I.A.) de alto desempenho exige grandes volumes de dados anotados e diversificados. Entretanto, em muitas áreas, esses dados não estão

disponíveis publicamente ou demandam investimentos significativos para coleta, curadoria e processamento. Em domínios sensíveis, como o educacional ou o corporativo, restrições de privacidade e questões éticas ampliam essa dificuldade, tornando inviável a criação de bases representativas em larga escala [3, 38].

A obtenção de conjuntos de dados adequados ao treinamento de CNNs representa um desafio multifacetado. Trabalhos como os de [34] e [39] destacam a complexidade associada ao uso de dados médicos, que frequentemente enfrentam barreiras relacionadas à privacidade e confidencialidade. Além das questões éticas e legais, limitações técnicas também influenciam a disponibilidade e o uso desses dados. [48] exploram os desafios da infraestrutura de dados, evidenciando como a ingestão e o armazenamento podem se tornar fatores limitantes no treinamento de modelos em larga escala, comprometendo inclusive o desempenho do sistema.

Essa visão é reforçada por [41], que abordam não apenas a quantidade necessária de dados, mas também a importância de sua qualidade e veracidade para garantir modelos confiáveis e robustos. Outro fator crítico é o custo e o tempo necessários para obter anotações manuais de alta qualidade. Conforme discutido por [12], o processo de rotulação é um gargalo no desenvolvimento de modelos de aprendizado profundo, pois demanda esforço humano intensivo e torna o processo caro e demorado. Essa problemática se intensifica em contextos de dados sensíveis, como apontado por [42], em que a coleta de amostras para casos raros enfrenta desafios adicionais, como a escassez geral de dados e a necessidade de especialistas para a anotação.

A escassez de dados anotados, aliada ao alto custo, às restrições éticas na coleta e aos gargalos de infraestrutura, limita o desenvolvimento e a validação de modelos robustos e generalizáveis. Em aplicações que demandam monitoramento comportamental, como análise de engajamento ou identificação de desvios em ambientes educacionais, a dificuldade em obter dados reais representativos restringe a construção e o teste de soluções eficientes [41].

A geração de dados sintéticos e a simulação de ambientes controlados surgem como alternativas estratégicas para mitigar essas limitações. Tais abordagens permitem criar cenários variados e controlados, preservar a privacidade, reduzir custos e viabilizar

experimentos replicáveis. Além disso, possibilitam a obtenção de situações raras ou extremas, difíceis de capturar em coletas reais [13, 18].

A utilização de simulações procedurais, aliada a modelos estatísticos e técnicas de visão computacional, oferece potencial para acelerar o desenvolvimento de soluções voltadas à análise de comportamento humano em ambientes multimodais. Essa abordagem viabiliza treinamentos de modelos de I.A. que combinam detecção de emoções, controle de presença e análise temporal sem exposição de dados pessoais, integrando-se a aplicações *web* para análise contínua de métricas de engajamento. No contexto do WebMedia, tal infraestrutura alinha-se a trabalhos como [25] e [32], contribuindo ao reduzir a dependência de coletas reais, mitigar questões de privacidade e permitir avaliação de modelos em condições experimentais variadas. Adicionalmente, viabiliza testes de mecânicas de *feedback* e cenários de interação, acelerando o desenvolvimento de sistemas em *User Experience* (UX), Computação Afetiva e multimídia educacional.

O objetivo principal do presente estudo é desenvolver um *framework* procedural para simulação de ambientes voltados ao treinamento de I.A., com foco no monitoramento de engajamento e na identificação de desvios comportamentais em contextos educacionais ou corporativos. A proposta permite criar cenários totalmente controlados para validar teorias e calibrar algoritmos, servindo como base para sistemas reais. O protótipo *Visage Track* é apresentado como prova de conceito, demonstrando que o *framework* pode reproduzir funcionalidades-chave de soluções consolidadas e gerar dados sintéticos adequados ao treinamento das mesmas arquiteturas.

2 TRABALHOS CORRELATOS

A escassez e o alto custo de obtenção de dados reais têm motivado o desenvolvimento de simuladores e *frameworks* para geração procedural de dados sintéticos em diferentes domínios.

Direção Autônoma. [24] propuseram simulador baseado em geração procedural para treinamento de agentes *end-to-end*, enquanto [11] desenvolveram o CARLA (*Car Learning to Act*), amplamente utilizado para avaliação de abordagens de condução autônoma. [1] expandiram a acessibilidade ao propor interface *no-code* para usuários não técnicos.

Robótica. [21] exploraram simulações procedurais para enriquecer comportamentos em robôs móveis. [26] apresentaram o *Isaac Gym*, que realiza simulação física e treinamento de políticas diretamente na GPU (*Graphics Processing Unit*), acelerando o processo em até 2–3 ordens de magnitude. [47] integraram o *Unity ML-Agents* (*Machine Learning Agents Toolkit*) para configuração simplificada de tarefas de locomoção. [19] desenvolveram o DISCOVERSE (*Discovery Environment for Real-to-Sim-to-Real Simulation and Evaluation*), combinando *Gaussian Splatting* e *MuJoCo* (*Multi-Joint dynamics with Contact*) para simulações hiper-realistas focadas no desafio *Real2Sim2Real*.

Ambientes Virtuais e Geração Procedural. [6] propuseram o ProcTHOR (*Procedural Task-oriented Household Object Rearrangement*), que gera proceduralmente ambientes domésticos para treinamento de agentes em tarefas de manipulação de objetos. [17] apresentaram

pipeline (fluxo de processamento) para geração sintética de imagens voltada à inspeção visual de superfícies metálicas. [2] utilizaram *Large Language Models* (LLMs, modelos de linguagem de grande escala) para projetar automaticamente funções de recompensa e otimizar geração procedural de conteúdo.

Lacuna Identificada. Os trabalhos apresentados demonstram avanços em geração procedural para domínios específicos (navegação, controle robótico, inspeção visual), porém não integram simultaneamente detecção de presença, reconhecimento emocional e monitoramento temporal de engajamento em ambientes simulados. Essa lacuna impede a geração controlada de dados sintéticos para análise comportamental humana, área que demanda simultaneidade de múltiplas modalidades (presença, emoção, atenção) e análise longitudinal. O presente trabalho preenche essa lacuna ao propor *framework* procedural que integra essas dimensões de forma unificada, permitindo geração escalável de dados sintéticos para treinamento de modelos de Inteligência Artificial (I.A.) voltados à análise de comportamento em contextos educacionais e corporativos, sem necessidade de exposição de dados pessoais reais.

3 METODOLOGIA

O *framework* proposto foi desenvolvido utilizando o motor *Unity* para criação do ambiente e modelagem gráfica com geração procedural [20, 31, 40]. Os principais componentes foram implementados em C# e integrados a um *backend* em *Golang* [10], responsável pela comunicação e processamento de dados. A arquitetura também incorporou CNNs para reconhecimento de expressões faciais em tempo real, adotando o modelo *MobileNetV2* com transferência de aprendizagem [4, 9]. O *MobileNetV2* foi selecionado devido à sua leveza e eficiência computacional, características que permitem execução em dispositivos com recursos limitados sem comprometer significativamente a precisão, sendo adequado para aplicações que demandam processamento em tempo real. O treinamento e a inferência das redes foram realizados em *Python*, utilizando *TensorFlow* e *Flask* [7, 16].

A interface multiplataforma foi desenvolvida em *Flutter* [46], assegurando compatibilidade tanto com dispositivos móveis quanto com sistemas *desktop*. O *framework* integra conceitos de ciência de dados e *aprendizado de máquina* (*Machine Learning* – ML) [27, 28, 30], fundamentando-se em padrões arquiteturais amplamente reconhecidos [14]. Para avaliar a eficiência e aplicabilidade da solução, foi desenvolvido um protótipo funcional, detalhado na Seção 3.3, que serviu como base para os testes e para a análise dos resultados obtidos.

3.1 Validação

A validação do *framework* foi conduzida por meio do protótipo denominado *Visage Track*, concebido para testar a aplicação prática da arquitetura proposta. O protótipo simula cenários controlados com indivíduos gerados proceduralmente (*Blobs*), permitindo mensurar o desempenho do sistema em tarefas como controle de presença, monitoramento de engajamento e detecção de desvios comportamentais.

Durante a execução, foram registradas métricas de processamento, volume de dados gerados e evolução das variáveis comportamentais ao longo do tempo. Essa abordagem possibilitou verificar

a robustez do *framework*, identificar limitações e comparar o comportamento do sistema com soluções de referência presentes na literatura.

3.2 Desenvolvimento do *Framework*

O *Framework* foi desenvolvido utilizando materiais modelados em 3D. Após a criação dos modelos, os seres foram denominados "Blobs" e projetados com características como cores, comportamentos e tamanhos distintos, conforme ilustrado na Figura 1. As características faciais, como o tamanho dos olhos e da boca, foram geradas proceduralmente, permitindo variações entre os Blobs. Os comportamentos definidos incluem: normal, triste, raiva, atento, dormindo, surpreso, feliz e ausência.

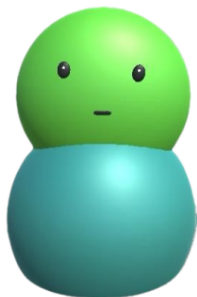


Figura 1: Blob em estado normal renderizado

Os comportamentos dos Blobs mudam em intervalos de 3 segundos dentro do ambiente simulado, permitindo observar variações de forma acelerada. A Figura 2 apresenta o ambiente renderizado.

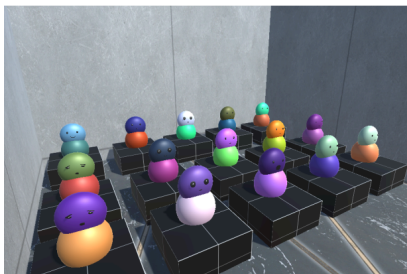
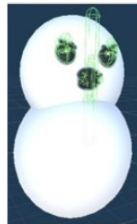


Figura 2: Ambiente Renderizado

3.2.1 Geração Procedural dos Blobs. O algoritmo de geração procedural introduziu variações sutis e únicas nos Blobs, partindo de um modelo base padrão. O algoritmo é responsável por ajustar parâmetros como tamanho da cabeça, do corpo e posições dos olhos e boca, utilizando multiplicadores aleatórios para garantir diversidade e consistência no design. Após ajustes, os Blobs são renderizados, assegurando unicidade e coesão visual, como mostrado na Figura 3.

3.2.2 Expressões faciais dos Blobs. As expressões faciais dos Blobs são determinadas por três variáveis principais: interesse, preguiça e comprometimento. Essas variáveis definem diferentes estados visuais, como "atento", "sorridente", "triste" ou "irritado", que podem ser observados alguns exemplos na Figura 4.

Modelo Base



Resultado Final

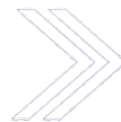


Figura 3: Processo de Geração do Blob

Quando o interesse do Blob está acima de 60%, há maior probabilidade de ele exibir emoções positivas, como sorrir ou expressar surpresa. Em contrapartida, valores abaixo de 35% aumentam as chances de emoções negativas, como tédio, sono ou tristeza. Níveis muito baixos de interesse e comprometimento podem levar ao desengajamento completo, caracterizado pela ausência do Blob no ambiente. O detalhamento do funcionamento dessa simulação pode ser encontrado na Seção 3.2.3.



Figura 4: Exemplos de expressões faciais dos Blobs

3.2.3 Simulação Estatística. Para modelagem do comportamento dos Blobs, desenvolveu-se um sistema de simulação baseado em geração procedural de valores pseudoaleatórios. Todos os atributos comportamentais — *laziness* (preguiça), *commitment* (comprometimento) e *interest* (interesse) — são inicializados no intervalo $[0, 100]$ e atualizados iterativamente para simular evolução ou regressão temporal.

Os scripts de geração procedural e treinamento do modelo estão disponibilizados publicamente [15].

Gerador Congruencial Linear. A atualização das variáveis utiliza um Gerador Congruencial Linear (*Linear Congruential Generator* – LCG) [22], definido pela recorrência apresentada na eq. (1):

$$X_{n+1} = (aX_n + c) \bmod m \quad (1)$$

onde X_n representa o estado atual do gerador, X_{n+1} o estado subsequente, a o multiplicador, c o incremento, e m o módulo. Os parâmetros adotados seguem a recomendação de KNUTH para sistemas de 32 bits, garantindo período máximo m conforme o teorema de Hull-Dobell [22], conforme especificado na Tabela 1.

Os valores gerados pela eq. (1) são inteiros no intervalo $[0, m-1]$ e são normalizados para o intervalo operacional $[0, 100]$ através da

Tabela 1: Parâmetros do Gerador Congruencial Linear utilizado.

Parâmetro	Símbolo	Valor
Multiplicador	a	1.664.525
Incremento	c	1
Módulo	m	2^{32}
Semente inicial	X_0	42

transformação linear apresentada na eq. (2):

$$x = \left(\frac{X_{n+1}}{m} \right) \times 100 \quad (2)$$

onde a divisão por m produz valores uniformemente distribuídos no intervalo $[0, 1)$, subseqüentemente escalonados para $[0, 100)$.

Taxa Diária de Redução da Preguiça. Para modelar a dinâmica de redução da preguiça ao longo do tempo, define-se a TDRP (*Taxa Diária de Redução da Preguiça*) conforme eq. (3):

$$\text{TDRP} = \frac{\text{commitment}}{2} + \left(\frac{\text{laziness}}{32 - \text{daysCount}} \right) \quad (3)$$

onde $\text{daysCount} \in [1, 31]$ representa o contador de dias transcorridos no ciclo de simulação. A restrição do intervalo garante estabilidade numérica e evita singularidades no denominador.

Dinâmica de Atualização do Interesse. A variável *interest* é atualizada mediante duas funções distintas, selecionadas com base no limiar $\text{commitment} = 45$. Para valores elevados de comprometimento ($\text{commitment} > 45$), aplica-se eq. (4):

$$\Delta \text{interest} = \left(\frac{5}{1 + \text{laziness}} \right) \cdot \left(\frac{\text{interest} \cdot \text{TDRP}}{100} \right) \quad (4)$$

Para comprometimento reduzido ($\text{commitment} \leq 45$), utiliza-se eq. (5):

$$\Delta \text{interest} = \left(\frac{\text{laziness}}{10} \right) \cdot \left(\frac{\text{interest} \cdot \text{TDRP}}{1000} \right) \quad (5)$$

O valor atualizado é então calculado como:

$$\text{interest}_{n+1} = \min(100, \max(0, \text{interest}_n + \Delta \text{interest})) \quad (6)$$

garantindo que a variável permaneça no intervalo válido $[0, 100]$.

O sistema de simulação opera com três componentes parametrizados. O Gerador Congruencial Linear, definido pela eq. (1), utiliza os parâmetros especificados na Tabela 1. O mecanismo de Decaimento Temporal com Reforço Periódico (*Temporal Decay with Reinforcement Probability* – TDRP), formalizado pela eq. (3), emprega divisor de *commitment* igual a 2, constante temporal de 32 unidades, e intervalo de *daysCount* no domínio $[1, 31]$. As equações de atualização eqs. (4) and (5) estabelecem limiar de *commitment* em 45, aplicando fator de escala 5 para regime de alto *commitment* (≥ 45) e fator 10 para regime de baixo *commitment* (< 45).

3.2.4 Modelo de Classificação Emocional. O modelo de classificação emocional implementado utiliza a arquitetura *MobileNetV2* [37], selecionada por sua eficiência computacional e adequação a dispositivos com recursos limitados. Esta arquitetura emprega convoluções separáveis em profundidade e blocos residuais invertidos, possibilitando inferências em tempo real sem comprometer a acurácia classificatória.

Dataset Sintético. O conjunto de dados compreende 5.000 imagens de Blobs geradas proceduralmente, apresentando variações sistemáticas em expressões faciais, paleta cromática e características morfológicas. A diversidade amostral visa mitigar problemas de sobreajuste (*overfitting*) e garantir generalização adequada do modelo. A Tabela 2 apresenta a distribuição das classes emocionais no dataset.

Tabela 2: Distribuição de classes emocionais no dataset sintético ($n = 5.000$).

Classe Emocional	Quantidade	Percentual (%)
Raiva	821	16,42
Triste	736	14,72
Normal	673	13,46
Feliz	602	12,04
Faltou	587	11,74
Surpreso	554	11,08
Entediado	501	10,02
Atento	319	6,38
Sonolento	207	4,14
Total	5.000	100,00

A distribuição espacial dos Blobs nas fileiras da grade não apresenta correlação significativa com estados emocionais específicos, dado que o posicionamento geométrico não influencia diretamente a expressão facial atribuída. Consequentemente, a distribuição por fileira mantém proporcionalidade estatística equivalente à distribuição global, com variações estocásticas inerentes ao processo de geração procedural. Embora a aleatoriedade introduza variabilidade não determinística, os desvios permanecem dentro de limites aceitáveis que não comprometem análises quantitativas subseqüentes.

Taxonomia Emocional. A Figura 5 ilustra as oito classes emocionais reconhecidas pelo modelo. O sistema implementa classificação multiclasse com agregação bifuncional, onde as emoções são categorizadas em valências afetivas positivas (feliz, surpreso, atento) e negativas (raiva, triste, entediado, sonolento), além da classe neutra (normal) que representa ausência de ativação emocional significativa.

O pipeline de inferência processa cada quadro capturado, classificando individualmente todos os Blobs visíveis. Os valores de ativação emocional atribuídos a cada expressão são agregados mediante função somatória, produzindo métrica unificada de engajamento coletivo. Esta abordagem permite quantificar o estado emocional global da população de Blobs em função temporal, subsidiando análises comportamentais ao longo do período observacional.

3.2.5 Transformação dos dados. O engajamento total (E_t) é calculado pela soma ponderada dos estados emocionais de todos os

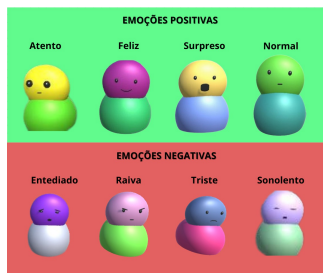


Figura 5: Taxonomia de expressões faciais para classificação emocional dos Blobs.

Blobs, incluindo ausências como estado negativo. A regressão linear foi selecionada por sua eficiência demonstrada e interpretabilidade superior: [45] evidenciaram desempenho comparável entre modelos lineares e não-lineares complexos (Pearson $r = 0,50$ vs. *boosting* $r = 0,52$), com vantagens em transparência analítica. A cada decêndio, dados de humor agregados geram pontos para construção da curva de regressão, identificando padrões evolutivos e correlações entre interações virtuais e respostas comportamentais coletivas. Esta abordagem mantém-se válida para modelagem de engajamento emocional em contextos educacionais onde interpretabilidade é fundamental [8].

Cada estado comportamental contribui diferenciadamente no cálculo: emoções como atenção e surpresa incrementam significativamente o engajamento, enquanto dormência e tristeza reduzem o valor final; estados neutros adicionam incremento positivo residual. Esses valores são somados e normalizados pelo número total de Blobs, resultando no engajamento médio utilizado na regressão. O processo, ilustrado na Figura 6, demonstra a classificação de expressões faciais e atribuição de valores correspondentes aplicados à fórmula de engajamento.

Os valores resultantes são confinados ao intervalo $[-50, 50]$, assegurando controle estatístico adequado para análise de regressão linear subsequente. Esta restrição normaliza a escala de mensuração, facilitando comparabilidade temporal e interpretação de tendências comportamentais.

Reconhecimento de Emoções

$$f(x) = \frac{[\text{blob_1}(\text{raiva}) + \text{blob_2}(\text{entediado}) + \text{blob_3}(x) \dots]}{\text{Quantidade Total de Blobs}}$$

Figura 6: Fórmula para cálculo do engajamento total como variável dependente em regressão linear

Paralelamente, realiza-se análise individual de cada participante para identificar desvios comportamentais ou questões que demandem intervenção. Diferentemente da análise coletiva, esta abordagem não emprega regressão linear, gerando notificações automáticas quando comportamentos alarmantes são detectados, viabilizando ações corretivas imediatas.

3.3 Desenvolvimento do Protótipo

3.3.1 Tecnologias Utilizadas. Durante o desenvolvimento, foram empregadas diversas linguagens de programação e tecnologias para atender aos objetivos do projeto. A linguagem C# foi utilizada devido à sua integração nativa com a plataforma Unity [20], facilitando a implementação de elementos gráficos e a lógica de interação no ambiente 3D.

O *backend* foi implementado em Go [10], escolhido por sua alta performance, suporte nativo à concorrência e pela familiaridade dos autores com a linguagem, o que favoreceu um desenvolvimento mais eficiente.

A linguagem Python foi adotada para as demandas relacionadas à I.A., integrando-se diretamente ao *framework* TensorFlow [7] para o treinamento e inferência de modelos. O *framework* Flask [16] foi empregado para disponibilizar serviços leves e de rápida configuração, permitindo a comunicação entre os módulos do sistema. Para a interface de usuário, utilizou-se a linguagem Dart em conjunto com o *framework* Flutter [46]. Embora a versão inicial fosse voltada apenas para a Web, a escolha pelo Flutter foi estratégica, considerando sua capacidade multiplataforma, o que possibilita futura portabilidade para dispositivos móveis e *desktop* sem necessidade de grandes reescritas no código.

4 RESULTADOS E DISCUSSÃO

Nesta seção, são apresentados os resultados obtidos com o uso do protótipo, discutindo-se sua eficácia e limitações. Além disso, os resultados são comparados com abordagens e trabalhos da literatura, destacando o potencial do *framework* proposto e identificando oportunidades para melhorias e extensões futuras.

4.1 Resultados do Framework Desenvolvido

O *framework* desenvolvido demonstrou a capacidade de simular diversos ambientes gerados proceduralmente. Para avaliar sua eficiência e a consistência dos dados produzidos, na Seção 4.1.1 foi realizada uma validação utilizando dados coletados pelo protótipo Visage Track, permitindo observar o comportamento dos dados ao longo de simulações extensas.

4.1.1 Análise de Múltiplas Simulações. Foram conduzidas quatro simulações com diferentes configurações paramétricas para avaliar a sensibilidade do *framework* às condições iniciais. A Figura 7 apresenta as séries temporais de engajamento obtidas, permitindo análise comparativa dos regimes dinâmicos emergentes.

Observa-se que a Simulação A ($T = 350$ dias) convergiu para regime estacionário, caracterizado por flutuações de baixa amplitude em torno de valor médio aproximadamente constante, indicando balanço entre termos de decaimento (*laziness*) e reforço (*commitment*). As Simulações B e D ($T = 50$ dias) exibiram trajetórias monotonicamente crescentes, porém com taxas de variação distintas: enquanto

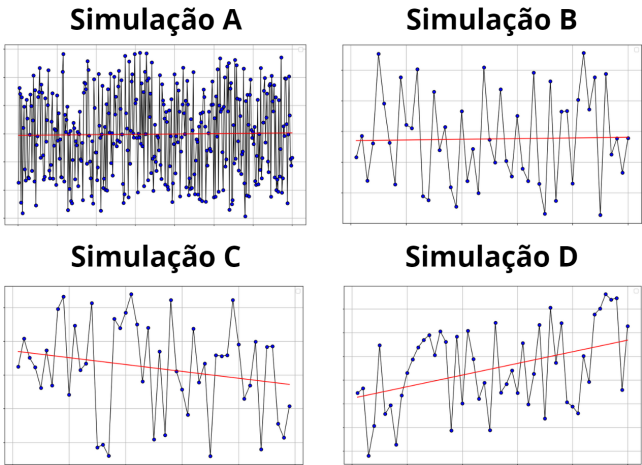


Figura 7: Evolução temporal do engajamento em quatro cenários simulados com diferentes configurações paramétricas.

B apresentou gradiente moderado, D demonstrou crescimento acentuado. Essa diferença sugere sensibilidade paramétrica do sistema às condições iniciais de *interest* e *commitment*, amplificadas pelo termo multiplicativo na eq. (4). A Simulação C manifestou decaimento monotônico, comportamento coerente com dominância do termo de *laziness* na eq. (5) sob regime de baixo *commitment* (< 45).

A análise comparativa revela três regimes distintos: (i) estabilidade dinâmica com oscilações limitadas, (ii) crescimento sustentado com taxas parametricamente moduladas, e (iii) decaimento sistemático. Essa diversidade de comportamentos valida a capacidade do *framework* de reproduzir fenômenos análogos a dinâmicas de engajamento em contextos educacionais, onde estagnação, progressão e regressão coexistem sob diferentes condições motivacionais.

4.1.2 Avaliação do Modelo de I.A. O modelo apresentou alta precisão para o controle de presença e reconhecimento de expressões faciais dos Blobs, com desempenho notável nas três primeiras fileiras. No entanto, testes com fileiras mais distantes revelaram uma queda significativa na precisão devido à distância e ao ângulo de captura, como detalhado na Tabela 3.

Fileira	Taxa de Acerto (%)
Primeira Fileira	98,01%
Segunda Fileira	96,32%
Terceira Fileira	92,78%

Tabela 3: Precisão do modelo

4.1.3 Avaliação da Capacidade de Generalização. Variações nos parâmetros dos Blobs, como tamanho do corpo e da cabeça, mostraram que a precisão diminui à medida que as alterações aumentam, conforme descrito na Tabela 4. Pequenas variações (0,3 e 0,5) mantiveram boa precisão, mas valores maiores, como 1, reduziram significativamente o desempenho, principalmente na terceira fileira.

Fileira	Variação de 0,3	Variação de 0,5	Variação de 1
Primeira Fileira	95,32%	91,21%	77,00%
Segunda Fileira	92,01%	86,13%	61,92%
Terceira Fileira	89,32%	76,21%	43,09%

Tabela 4: Precisão do modelo por diferentes valores de variação

Os resultados demonstraram que o sistema simula com eficácia diferentes padrões de comportamento e identifica desvios, destacando momentos críticos. Isso contribui para intervenções direcionadas em ambientes reais, como salas de aula, ajustando estratégias pedagógicas e melhorando o engajamento dos indivíduos.

O modelo demonstrou bom desempenho em ambientes controlados, mas apresentou limitações em cenários de maior complexidade, como fileiras distantes e variações extremas nos parâmetros dos Blobs, conforme apontado pela Tabela 4. Estratégias como *data augmentation*, maior diversidade nos dados de treinamento e uso de modelos mais robustos podem melhorar a generalização.

A Figura 8 apresenta um exemplo de detecção dos Blobs pelo modelo desenvolvido, ilustrando o funcionamento no ambiente simulado criado.



Figura 8: Funcionamento do Modelo no Ambiente Simulado

4.1.4 Análise de Desvios Comportamentais. Os dados individuais dos Blobs foram analisados para detectar desvios comportamentais, gerando notificações específicas Figura 9. Esse recurso auxilia na identificação de problemas e permite ações preventivas.



Figura 9: Relatório de desvios comportamentais

Os relatórios individuais são gerados continuamente durante a simulação, operando em segundo plano. O usuário responsável pelo gerenciamento do ambiente ou turma recebe notificações com base nos padrões comportamentais observados em cada Blob nos dias anteriores. Caso seja detectado um rompimento significativo nos padrões comportamentais estabelecidos, o sistema envia um alerta imediato ao usuário.

4.1.5 *Comparação com trabalhos correlatos.* Os trabalhos descritos na Seção 2 concentram-se em domínios como direção autônoma, robótica e inspeção visual, não abordando de forma integrada análise de sentimentos, engajamento e comportamento individual. Outros estudos, como [29], exploram a geração procedural de vídeos hiper-realistas para detecção de ações raras; [43] utilizam ambientes virtuais gerados em *Unity* para estudos de fobia em VR; [5] simulam interações em grupo com humanos virtuais fotorealistas; [33] criam agentes autônomos com comportamento social emergente; e [36] geram cenas com avatares 3D para aumentar a variabilidade de dados de ação humana. Embora mais próximos do estudo de emoções e interações humanas, esses trabalhos não integram presença, engajamento e reconhecimento de emoções em um único *framework* voltado à geração controlada de dados para análise comportamental.

A Tabela 5 sintetiza os trabalhos correlatos em geração procedural, organizando-os por domínio de aplicação e foco principal. Observa-se que a maior parte das iniciativas concentra-se em áreas específicas, como direção autônoma, robótica e visão computacional para análise de ações, com simulações voltadas a cenários técnicos ou de navegação, mas sem integração direta com métricas de presença, engajamento e reconhecimento de emoções. Trabalhos como [43], [33], [29], [5] e [36] aproximam-se mais do estudo de interações humanas, porém ainda mantêm escopo restrito, seja ao estudo de fobias, comportamentos emergentes ou detecção de ações específicas.

Tabela 5: Classificação dos trabalhos correlatos por domínio de aplicação

Trabalho	Domínio
[24]	Direção autônoma
[11]	Direção autônoma
[1]	Direção autônoma
[23]	Direção autônoma
[6]	Ambientes virtuais
[43]	Ambientes virtuais
[33]	Ambientes virtuais
[21]	Robótica
[2]	Robótica
[47]	Robótica
[26]	Robótica
[19]	Robótica
[29]	Visão computacional
[5]	Visão computacional
[36]	Visão computacional
Presente trabalho	Análise de comportamento humano

O presente trabalho diferencia-se por unir múltiplos aspectos de análise comportamental em um único *framework* procedural adaptável, permitindo a criação de ambientes virtuais com controle total das variáveis e geração de dados sintéticos voltados simultaneamente à presença, engajamento e reconhecimento de emoções. Essa abordagem amplia a aplicabilidade das simulações, atendendo tanto a estudos de comportamento humano em contextos controlados quanto ao treinamento e validação de modelos de inteligência artificial de forma escalável e ética.

O *framework* proposto diferencia-se por oferecer uma base procedural adaptável para criar ambientes virtuais voltados a interações humanas, permitindo controle total das variáveis e geração de dados específicos para treinar e avaliar modelos de inteligência artificial

nesse contexto. O protótipo *Visage Track* demonstra essa aplicabilidade, servindo como gerador de dados sintéticos para cenários que carecem de conjuntos de dados controlados e escaláveis.

4.1.6 *Limitações do Framework.* O *framework* adota representações visuais simplificadas (*Blobs*) que priorizam a qualidade dos dados emocionais e comportamentais em detrimento da complexidade gráfica. Embora essa escolha seja deliberada para focar em transições emocionais e padrões de engajamento, reconhece-se que a natureza simplista dos *Blobs* limita a generalização para contextos visualmente complexos com variações realistas de iluminação, textura e morfologia facial.

Para mitigar o *domain gap* entre dados sintéticos e reais, propõe-se como trabalho futuro: (i) *domain randomization* com variações procedurais de iluminação, ângulos de câmera e ruído visual; e (ii) *domain adaptation* via *fine-tuning* com dados reais ou redes adversariais de domínio (DANN), visando melhorar a transferibilidade para aplicações práticas em cenários educacionais e corporativos.

4.2 Resultados do Protótipo Desenvolvido

4.2.1 *Estrutura do Software.* O sistema inicia na Simulação em *Unity*, que gera dados do ambiente virtual. Esses dados são processados pelo Servidor *Flask*, responsável pela inferência de I.A. e cálculos de regressão linear, e posteriormente enviados ao Servidor *Golang*, que gerencia a autenticação e integra os módulos de simulação. As informações finais são exibidas na Interface Web, permitindo análises, gráficos e resultados detalhados. A Figura 10 apresenta o fluxo do sistema.

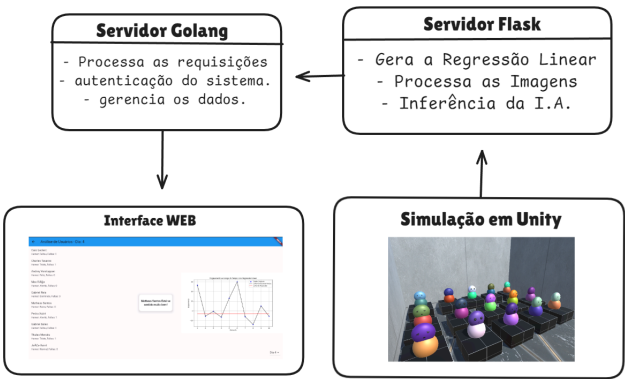


Figura 10: Fluxograma simplificado de funcionamento do protótipo

4.2.2 *Visualização de Dados.* A interface web apresenta, em tempo real, informações como identificação, estados emocionais e presença dos indivíduos simulados (*Blobs*). Gráficos de regressão linear são atualizados a cada 10 dias de simulação, permitindo o acompanhamento da evolução do engajamento e a detecção de tendências. O sistema também emite alertas automáticos quando há desvios comportamentais relevantes (por exemplo, tristeza prolongada), viabilizando intervenções imediatas no ambiente virtual.

A infraestrutura foi concebida para integração direta com aplicações web, de modo a apoiar visualização de resultados e análise

temporal de engajamento. Embora o ambiente seja simulado, o objetivo é a geração de *datasets* e métricas para treinar e validar modelos de I.A. voltados a emoções e comportamento humano. Assim, a proposta dialoga com investigações em Experiência do Usuário (*User Experience* — UX) e Computação Afetiva, oferecendo um recurso controlado e escalável para estudos de interação e comportamento.

4.2.3 Limitações do Protótipo. O modelo de engajamento baseado em regressão linear apresenta limitações significativas ao assumir tendências constantes de progressão ou declínio, quando na prática o engajamento humano exhibe padrões oscilatórios e não-lineares. Para superar essa simplificação, propõe-se a adoção de modelos não-lineares, como análise de séries temporais com autocorrelação ou redes neurais recorrentes (LSTM), capazes de capturar flutuações momentâneas de atenção, fadiga intermitente e variações contextuais que caracterizam a dinâmica real do engajamento em ambientes educacionais e corporativos.

4.3 Validação

Para avaliar os resultados e o potencial do protótipo, foi realizada uma análise comparativa entre as funcionalidades oferecidas pelo *Visage Track* e dois trabalhos relevantes na literatura: o modelo de [44], referência na análise de emoções no contexto educacional, e o de [35], voltado à detecção precoce de desengajamento estudantil por meio de visão computacional.

A Tabela 6 apresenta a comparação direta, considerando oito critérios principais: *Emoções*, *Presença*, *API*, *Modelo (I.A.)*, *Interface*, *Tempo Real*, *Emoções Subjetivas* e *Temporal*. Esses critérios foram classificados em quatro níveis: *Melhor* (desempenho superior), *Parcial* (vantagem leve), *Igual* (desempenho semelhante) e *Pior* (desempenho inferior).

Tabela 6: Comparação entre o *Visage Track*, [44] e [35]

Critério	<i>Visage Track</i>	Werlang	Pillai
Emoções	Parcial	Melhor	Melhor
Presença	Melhor	Pior	Pior
API	Parcial	Melhor	Melhor
Modelo (I.A.)	Igual	Igual	Melhor
Interface	Parcial	Melhor	Melhor
Tempo Real	Melhor	Pior	Pior
Emoções Subjetivas	Pior	Melhor	Melhor
Temporal	Melhor	Pior	Pior

Os resultados evidenciam que o *Visage Track* apresenta vantagens na detecção de presença e no acompanhamento temporal de interações, funcionalidades ausentes em [44] e [35], possibilitando automação de chamadas e análises longitudinais de engajamento. Contudo, ambos os trabalhos comparados superaram o protótipo na análise de emoções complexas e subjetivas, como confusão, frustração, engajamento e desinteresse, utilizando *datasets* especializados e arquiteturas mais avançadas. Quanto à integração com sistemas externos, [44] e [35] dispõem de APIs mais maduras, enquanto o *Visage Track* oferece base parcial com potencial de expansão.

Cabe destacar que [44] e [35] são sistemas completos para uso direto, enquanto o *Visage Track* constitui prova de conceito para validação do *framework* proposto. Apesar dessa diferença de escopo, o protótipo demonstrou desempenho relevante em funcionalidades-chave, evidenciando aplicabilidade prática e potencial para gerar

dados sintéticos capazes de treinar as mesmas arquiteturas de IA empregadas nos sistemas comparados.

5 CONCLUSÃO

Os objetivos desta pesquisa foram alcançados mediante o desenvolvimento de um *framework* procedural para geração de dados sintéticos voltado ao treinamento de modelos de I.A. em contextos de monitoramento de engajamento e identificação de desvios comportamentais. O protótipo *Visage Track* validou a arquitetura proposta, demonstrando capacidade de reproduzir cenários controlados com variações emocionais de agentes simulados, estabelecendo uma base experimental escalável e eticamente viável para estudos comportamentais sem dependência de dados reais.

A principal contribuição reside na integração de métricas multimodais (presença, engajamento e reconhecimento emocional) em arquitetura procedural adaptável. Diferentemente de abordagens correlatas que abordam aspectos isolados, o *framework* proposto permite simulação com controle total de variáveis, viabilizando aplicações em computação afetiva, análise comportamental e pesquisas de UX em ambientes simulados.

As limitações observadas incluem: (i) simplicidade das representações visuais dos *Blobs*, restringindo generalização para contextos reais com maior complexidade morfológica e comportamental; (ii) modelo de regressão linear que não captura padrões oscilatórios e não-lineares característicos do engajamento humano, assumindo tendências constantes inadequadas para dinâmicas temporais efetivas; (iii) redução na precisão da arquitetura *MobileNetV2* sob condições fora do domínio de treinamento (variações de distância e características morfológicas extremas), evidenciando limitações de generalização para cenários complexos.

Como trabalhos futuros, propõe-se a inclusão de dados simulados com maior diversidade morfológica e comportamental, evoluindo a geração procedural dos *Blobs* para representações mais realistas mediante técnicas de *rigging* facial avançado que facilitem a transição para validação com dados reais via *sim-to-real transfer*. No contexto de aplicações práticas em sistemas multimídia, o *framework* apresenta potencial de integração com plataformas de *streaming* educacional e *Learning Management Systems* (LMS), viabilizando análise de engajamento em tempo real e correlação de métricas comportamentais com desempenho acadêmico. Recomenda-se investigar a aplicabilidade em domínios adjacentes, como treinamento corporativo, ambientes de realidade virtual (VR) e aumentada (AR), e sistemas de recomendação sensíveis ao contexto emocional.

REFERÊNCIAS

- [1] Ahmed Abouelazm, Mohammad Mahmoud, Conrad Walter, Oleksandr Shchetsura, Erne Hussong, Helen Gremmelmaier, and J. Marius Zöllner. 2025. Bridging Simulation and Usability: A User-Friendly Framework for Scenario Generation in CARLA. arXiv:2507.19883 [cs.RO] <https://arxiv.org/abs/2507.19883>
- [2] In-Chang Baek, Tae-Hwa Park, Jin-Ha Noh, Cheong-Mok Bae, and Kyung-Joong Kim. 2024. ChatPCG: Large Language Model-Driven Reward Design for Procedural Content Generation. *arXiv preprint* (2024). Available at <https://arxiv.org/abs/2406.11875>.
- [3] L. Brigato and L. Iocchi. 2020. A Close Look at Deep Learning with Small Data. arXiv:2003.12843 [cs.LG] Available at <https://arxiv.org/abs/2003.12843>.
- [4] Felipe Zago Canal. 2024. *Método para reconhecimento em tempo real de expressões faciais em grupos utilizando redes neurais convolucionais*. Dissertação (Mestrado). Universidade Federal de Santa Catarina, Araranguá, SC. Advisor(s) Pozzebon, Eliane. Available at <https://repositorio.ufsc.br/handle/123456789/254899>.

- [5] Che-Jui Chang, Danrui Li, Deep Patel, Parth Goel, Honglu Zhou, Seonghyeon Moon, Samuel S. Sohn, Sejong Yoon, Vladimir Pavlovic, and Mubbasir Kapadia. 2024. M3Act: Learning from Synthetic Human Group Activities. *arXiv:2306.16772* [cs.CV]. <https://arxiv.org/abs/2306.16772>
- [6] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Jordi Salvador, Kiana Ehsani, Winson Han, Eric Kolve, Ali Farhadi, Aniruddha Kembhavi, and Roozbeh Mottaghi. 2022. Procthor: Large-Scale Embodied AI Using Procedural Generation. *arXiv preprint* (2022). Available at <https://procthor.allenai.org>.
- [7] TensorFlow Developers. 2022. TensorFlow. *Zenodo* (2022). Available at <https://zenodo.org/record/4724015>.
- [8] M. Ali Akber DEWAN, Mahub MURSHED, and Fuhua LIN. 2019. Engagement detection in online learning: a review. *Smart Learning Environments* 6, 1 (jan 2019), 1. <https://doi.org/10.1186/s40561-018-0080-z>
- [9] Ke Dong, Chengjie Zhou, Yihan Ruan, and Yuzhi Li. 2020. MobileNetV2 model for image classification. In *2020 2nd International Conference on Information Technology and Computer Application (ITCA)*. IEEE, 476–480. DOI: 10.1109/ITCA52113.2020.00106. Available at <https://doi.org/10.1109/ITCA52113.2020.00106>.
- [10] Alan AA Donovan and Brian W Kernighan. 2015. *The Go programming language*. Addison-Wesley Professional.
- [11] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. 2017. CARLA: An Open Urban Driving Simulator. Available at <https://arxiv.org/abs/1711.03938>.
- [12] Zeyad Emam, Andrew Kondrich, Sasha Harrison, Felix Lau, Yushi Wang, Aerin Kim, and Elliot Branson. 2021. On The State of Data In Computer Vision: Human Annotations Remain Indispensable for Developing Deep Learning Models. *arXiv:2108.00114* [cs.CV]. Available at <https://arxiv.org/abs/2108.00114>.
- [13] Sophie Feinauer, Lea Schuller, Irene Groh, Lynn Huestegge, and Tibor Petzoldt. 2022. O potencial da gamificação para a educação do usuário em automação de direção parcial e condicional: um estudo com simulador de direção. *Transportation Research Part F: Traffic Psychology and Behaviour* 90 (2022), 252–268. <https://doi.org/10.1016/j.trf.2022.08.009>
- [14] Martin Fowler. 2002. *Patterns of Enterprise Application Architecture*. Addison-Wesley Professional, Boston, MA. ISBN: 978-0321127426.
- [15] Jäder Gonçalves. 2025. VisageTrack Project. GitHub. <https://github.com/visagetrack-project> Acesso em: 8 out. 2025.
- [16] Miguel Grinberg. 2018. *Flask web development*. "O'Reilly Media, Inc."
- [17] Pierre Gutierrez, Maria Luschkova, Antoine Cordier, Mustafa Shukor, Mona Schappert, and Tim Dahmen. 2021. Synthetic training data generation for deep learning based quality inspection. In *Fifteenth International Conference on Quality Control by Artificial Vision*, Takashi Komuro and Tsuyoshi Shimizu (Eds.), Vol. 11794. International Society for Optics and Photonics, SPIE, 1179403. DOI: 10.1117/12.2586824. Available at <https://doi.org/10.1117/12.2586824>.
- [18] Aravinda Jatavallabha. 2024. O piloto automático da Tesla: ética e tragédia. *arXiv preprint arXiv:2409.17380*. <https://arxiv.org/abs/2409.17380> Licença: CC BY 4.0.
- [19] Yufei Jia, Guangyu Wang, Yuhang Dong, Junzhe Wu, Yupei Zeng, Haonan Lin, Zifan Wang, Haizhou Ge, Weibin Gu, Kairui Ding, Zike Yan, Yunjie Cheng, Yue Li, Ziming Wang, Chuxuan Li, Wei Sui, Lu Shi, Guanzhong Tian, Ruqi Huang, and Guyue Zhou. 2025. DISCOVERSE: Efficient Robot Simulation in Complex High-Fidelity Environments. *arXiv:2507.21981* [cs.RO]. <https://arxiv.org/abs/2507.21981>
- [20] Arthur Juliani, Vincent-Pierre Berges, Ervin Teng, Andrew Cohen, Jonathan Harper, Chris Elion, Chris Goy, Yuan Gao, Hunter Henry, Marwan Mattar, and Danny Lange. 2020. Unity: A General Platform for Intelligent Agents. *arXiv preprint* (2020). Available at <https://arxiv.org/abs/1809.02627v2>.
- [21] Victor A Kich, Jair A Bottega, Raul Steinmetz, Ricardo B Grando, Ayanori Yorozu, and Akihisa Ohya. 2024. Advancing Behavior Generation in Mobile Robotics through High-Fidelity Procedural Simulations. *arXiv preprint* (2024). Available at <https://arxiv.org/abs/2405.16818>.
- [22] Donald Ervin KNUTH. 1997. *The art of computer programming* (3 ed.). Vol. 2. Addison-Wesley, Boston. Seção 3.2.1: The Linear Congruential Method.
- [23] Quanyi Li, Zhenghao Peng, Lan Feng, Qihang Zhang, Zhenghai Xue, and Bolei Zhou. 2022. MetaDrive: Composing Diverse Driving Scenarios for Generalizable Reinforcement Learning. *arXiv:2109.12674* [cs.LG]. <https://arxiv.org/abs/2109.12674>
- [24] Quanyi Li, Zhenghao Peng, Qihang Zhang, Cong Qiu, Chunxiao Liu, and Bolei Zhou. 2020. Improving the Generalization of End-to-End Driving through Procedural Generation. *arXiv preprint* (2020). Available at <https://arxiv.org/abs/2012.13681>.
- [25] Paulo Victor Borges Oliveira Lima, Djeferson Maranhão, and Carlos De Salles Soares. 2023. Automatic Emotion Detection in the Learning of Algorithms. In *Anais do XXIX Simpósio Brasileiro de Sistemas Multimídia e Web (WebMedia 2023)*. Sociedade Brasileira de Computação, Ribeirão Preto, SP, 56–64.
- [26] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and Gavriel State. 2021. Isaac Gym: High Performance GPU-Based Physics Simulation For Robot Learning. *arXiv:2108.10470* [cs.RO]. <https://arxiv.org/abs/2108.10470>
- [27] Wes McKinney. 2018. *Python para análise de dados: Tratamento de dados com Pandas, NumPy e IPython*. Novatec Editora.
- [28] Nilo Ney Coutinho Menezes. 2021. *Introdução à programação com Python* (3ª ed.). Novatec Editora.
- [29] David Mulero-Pérez, Manuel Benavent-Lledo, Jose Garcia-Rodriguez, and Markus Vincze. 2025. Unrealgensyn: a framework for generating synthetic videos of Unfrequent human events. *Virtual Reality* 29, 2 (2025), 76. <https://doi.org/10.1007/s10055-025-01146-9>
- [30] Andreas C Müller and Sarah Guido. 2016. *Introduction to machine learning with Python: a guide for data scientists*. "O'Reilly Media, Inc."
- [31] Alex Okita. 2019. *Learning C# programming with Unity 3D*. AK Peters/CRC Press.
- [32] Paula T. Palomino, Lennart Nacke, and Seiji Isotani. 2023. Gamificação Narrativa para Engajamento e Personalização: Redefinindo a Experiência do Aprendizado Digital. In *Anais do XXIX Simpósio Brasileiro de Sistemas Multimídia e Web (WebMedia 2023) – Concurso de Teses e Dissertações*. Sociedade Brasileira de Computação, Ribeirão Preto, SP, 27–30. https://doi.org/10.5753/webmedia_estendido.2023.233396
- [33] Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative Agents: Interactive Simulacra of Human Behavior. *arXiv:2304.03442* [cs.HC]. <https://arxiv.org/abs/2304.03442>
- [34] Yang Peng, Ming Zhang, Fangqiang Yu, Jinglin Xu, and Shang Gao. 2020. Digital twin hospital buildings: an exemplary case study through continuous lifecycle integration. *Advances in Civil Engineering* 2020, 1 (2020), 8846667. DOI: 10.1155/2020/8846667. Available at <https://doi.org/10.1155/2020/8846667>.
- [35] Aravind Sasidharan Pillai. 2022. Student Engagement Detection in Classrooms through Computer Vision and Deep Learning: A Novel Approach Using YOLOv4. *Sage Science Review of Educational Technology* 5, 1 (2022), 87–97. Available at <https://journals.sagescience.org/index.php/ssret/article/view/144>.
- [36] Ana Romero, Pedro Carvalho, Luis Córte-Real, and Américo Pereira. 2023. Synthesizing Human Activity for Data Generation. *Journal of Imaging* 9, 10 (2023). <https://doi.org/10.3390/jimaging9100204>
- [37] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2019. MobileNetV2: Inverted Residuals and Linear Bottlenecks. *arXiv:1801.04381* [cs.CV]. <https://arxiv.org/abs/1801.04381>
- [38] Ole Schmedemann, Melvin Baaß, Daniel Schoepflin, and Thorsten Schüppstühl. 2022. Procedural synthetic training data generation for AI-based defect detection in industrial surface inspection. *Procedia CIRP* 107 (2022), 1101–1106. Leading manufacturing systems transformation – Proceedings of the 55th CIRP Conference on Manufacturing Systems 2022. DOI: 10.1016/j.procir.2022.05.115.
- [39] Maxime Serresant, Hervé Delingette, Hubert Cochet, Pierre Jais, and Nicholas Ayache. 2021. Applications of artificial intelligence in cardiovascular imaging. *Nature Reviews Cardiology* 18, 8 (2021), 600–609. DOI: 10.1038/s41569-021-00544-y.
- [40] Noor Shaker, Julian Togelius, and Mark J. Nelson. 2016. *Procedural Content Generation in Games*. Springer. DOI: 10.1007/978-3-319-42716-4.
- [41] Daniel T. Speckhard, Tim Bechtel, Luca M. Ghiringhelli, Martin Kuban, Santiago Rigamonti, and Claudia Draxl. 2024. How big is Big Data? *arXiv:2405.11404* [stat.ML]. Available at <https://arxiv.org/abs/2405.11404>.
- [42] Vajira Thambawita, Pegah Salehi, Sajad Amouei Sheshkal, Steven A. Hicks, Hugo L. Hammer, Sravanthi Parasa, Thomas de Lange, Pål Halvorsen, and Michael A. Riegler. 2022. SinGAN-Seg: Synthetic training data generation for medical image segmentation. *PLOS ONE* 17, 5 (05 2022), 1–24. DOI: 10.1371/journal.pone.0267976. Available at <https://doi.org/10.1371/journal.pone.0267976>.
- [43] Kaitlyn Tracy, Lazaros Rafail Kouzelis, Rami Dari, and Ourania Spantidi. 2025. Plangen: A Procedural Generation Framework for Dynamic VR Research Environments. *Virtual Worlds* 4, 3 (2025). <https://doi.org/10.3390/virtualworlds4030033>
- [44] Pablo Werlang and Patrícia A. Jaques. 2023. Detecção por face de emoções de aprendizagem: abordagem baseada em redes neurais profundas e fluxo de emoções. *Revista Brasileira de Informática na Educação – RBIE* 31 (2023), 174–204. DOI: 10.5753/rbie.2023.2936. Available at <http://br-ie.org/pub/index.php/rbie>.
- [45] Jacob Whitehill, Zewelanji Serpell, Yi-Ching Lin, Aysa Foster, and Javier R. Movellan. 2014. The Faces of Engagement: Automatic Recognition of Student Engagement from Facial Expressions. *IEEE Transactions on Affective Computing* 5, 1 (2014), 86–98. <https://doi.org/10.1109/TAFFC.2014.2316163>
- [46] Eric Windmill. 2020. *Flutter in Action*. Simon and Schuster.
- [47] Linqi Ye, Rankun Li, Xiaowen Hu, Jiayi Li, Boyang Xing, Yan Peng, and Bin Liang. 2025. Unity RL Playground: A Versatile Reinforcement Learning Framework for Mobile Robots. *arXiv:2503.05146* [cs.RO]. <https://arxiv.org/abs/2503.05146>
- [48] Mark Zhao, Niket Agarwal, Aarti Basant, Buğra Gedik, Satadru Pan, Mustafa Ozdal, Rakesh Komuravelli, Jerry Pan, Tianshu Bao, Haowei Lu, Sundaram Narayanan, Jack Langman, Kevin Wilfong, Harsha Rastogi, Carole-Jean Wu, Christos Kozyrakis, and Parik Pol. 2022. Understanding data storage and ingestion for large-scale deep recommendation model training: industrial product. In *Proceedings of the 49th Annual International Symposium on Computer Architecture*. ACM, 1042–1057. DOI: 10.1145/3470496.3533044. Available at <http://dx.doi.org/10.1145/3470496.3533044>.