Robustness Analysis of Deep Reinforcement Learning Agents Under Visual Perturbations

Ana Lu Tan[1], **Arthur Monteiro Senra**[2], **Thiago Coelho**[3], **Daniel R. Figueiredo**[3]

[1]Instituto de Computação – Universidade Federal do Rio de Janeiro (UFRJ)

[2]Escola Politécnica – Universidade Federal do Rio de Janeiro (UFRJ)

[3]Programa de Engenharia de Sistemas e Computação – Universidade Federal do Rio de Janeiro (UFRJ)

analt@ic.ufrj.br, arthursenra.20231@poli.ufrj.br, tcoelho@cos.ufrj.br,

daniel@cos.ufrj.br

Abstract. *Reinforcement learning is one of the key paradigms of machine learning to solve problems where sequential decisions need to be made, such as robot and video game control. The emergence of reinforcement learning coupled to deep neural networks (*deep reinforcement learning*) allowed to solve problems of great complexity never before solved. In this context, one of the main architectures is Deep Q-Network (DQN), that generalizes the classic Q-learning algorithm using neural networks. However, the generalization of a trained model to other environments is still a challenge. This work evaluates the impact of different visual perturbations on a DQN agent in Pong, with additional experiments in Boxing and Space Invaders. Results indicate that the agent can tolerate certain disturbances, but its performance drops abruptly when the disturbances increase, indicating a high sensitivity to the disturbances. We also show that an agent trained in an environment with visual perturbations can have high performance but when being evaluated in the game without disturbances its performance is very low, indicating the fragility of DQN in learning to play.*

Resumo. *Aprendizado por reforço é um dos principais paradigmas de aprendizado de máquina para resolver problemas onde decisões sequenciais precisam ser tomadas, como o controle de robôs e jogos eletrônicos. O surgimento de aprendizado por reforço acoplado a redes neurais profundas (*deep reinforcement learning*) permitiu solucionar problemas de grande complexidade nunca antes resolvidos. Neste contexto, uma das principais arquiteturas é a Deep Q-Network (DQN), que generaliza o algoritmo clássico de Q-learning utilizando redes neurais. Entretanto, a generalização de um modelo treinado para outros ambientes ainda é um desafio. Este trabalho avalia o impacto de diferentes perturbações visuais em um agente DQN no jogo Pong, com experimentos adicionais em Boxing e Space Invaders. Resultados indicam que o agente consegue tolerar certas perturbações, mas seu desempenho cai abruptamente quando as perturbações aumentam, indicando uma alta sensibilidade às perturbações. Mostramos ainda que um agente treinado em um ambiente com perturbação pode ter alto desempenho, mas ao ser avaliado no jogo sem perturbações seu desempenho é muito baixo, indicando a fragilidade da DQN em aprender a jogar.*

1. Introdução

O paradigma de aprendizado por reforço constitui-se como um expoente na área do aprendizado de máquina para resolver problemas onde decisões sequenciais precisam ser tomadas, como o controle de robôs ou jogos eletrônicos [Sutton and Barto 2018]. A aplicação de redes neurais profundas em problemas de grande complexidade se mostrou uma verdadeira evolução, permitindo a solução de problemas nunca antes resolvidos [Mnih et al. 2015]. Neste contexto, uma das principais ideias é utilizar redes profundas para representar estados do sistema em baixa dimensão e aproximar o valor das ações para estas representações, generalizando o algoritmo clássico de Q-learning. Uma das principais arquiteturas nesta direção é a Deep Q-Network (DQN), sendo a primeira a ser proposta e amplamente avaliada no contexto de jogos eletrônicos [Mnih et al. 2015]. Essa capacidade de representar informações visuais em ambientes reais de alta dimensão levou a avanços expressivos nas áreas de robótica e controle de sistemas.

Contudo, mesmo com a superação do desempenho humano por agentes treinados em ambientes de simulação controlada e frequentemente determinísticos, um dos desafios desse paradigma é a generalização desses modelos para ambientes com perturbações visuais. Por exemplo, pequenas modificações nos pixels da tela que representa o estado do sistema que seriam facilmente negligenciadas por jogadores humanos (como flutuações de iluminação, ruídos térmicos de sensores ou a inserção de elementos de distração) podem degradar significativamente o desempenho das redes neurais profundas.

Entender como diferentes ruídos visuais influenciam o desempenho de algoritmos de aprendizado por reforço profundo é crucial para determinar a robustez e adaptabilidade dos modelos treinados, e também melhor compreender o que de fato esses modelos aprendem. Um ambiente para análise dos impactos de perturbações visuais no desempenho dos agentes são jogos de Atari, devido à sua simplicidade gráfica e jogabilidade. Nesse contexto, utilizando como ambiente o jogo Pong, propomos uma análise quantitativa e qualitativa do desempenho de uma DQN quando submetida a quatro perturbações visuais diferentes: ruído gaussiano, inclusão de elementos visuais estáticos, oclusão parcial e total da movimentação do adversário.

Para realizar este estudo de robustez, um agente é treinado utilizando DQN no jogo original, sem nenhuma perturbação visual, obtendo um desempenho excepcional no jogo. Em seguida, o agente treinado é submetido a jogos com perturbações visuais, nunca antes vistas na fase de treino. Resultados indicam que ruído gaussiano e inclusão de elementos estáticos degradam a qualidade do agente de forma abrupta: com pequenas perturbações o agente demonstra bom desempenho, que degrada drasticamente ao aumentar a intensidade ou quantidade das perturbações. A oclusão parcial e total da movimentação do adversário também possui um efeito parecido, reduzindo relativamente pouco o desempenho do agente com a oclusão parcial e drasticamente com a oclusão total. Além disso, foram avaliados em ambiente com ruído gaussiano agentes treinados em outros dois jogos: Boxing e Space Invaders, onde também foi observada uma degradação do desempenho do agente, indicando que este efeito não é exclusivo de um jogo em específico.

Um aspecto fundamental é a capacidade do agente de ser treinado e ter bom desempenho em ambiente com perturbações visuais. De fato, mostramos que isso é possível no caso mais desafiador: a oclusão total da movimentação do adversário. Entretanto, este mesmo agente possui um desempenho muito baixo quando avaliado no ambiente original,

sem perturbações visuais. Este exemplo demonstra a fragilidade do aprendizado sendo realizado pela DQN, demonstrando a necessidade de se explorar arquiteturas mais robustas para aprendizado por reforço.

O restante deste artigo está organizado da seguinte forma. Na seção 2 alguns conceitos importantes são apresentados juntamente com alguns trabalhos relacionados. A seção 3 apresenta a metodologia de avaliação descrevendo as perturbações visuais analisadas. Os resultados são apresentados na seção 4, e a seção 5 apresenta uma breve conclusão do artigo.

2. Conceitos e trabalhos relacionados

2.1. Conceitos Fundamentais

O aprendizado por reforço é um paradigma de aprendizado de máquina no qual um agente aprende a interagir com um ambiente guiado por recompensas. O agente é a entidade que toma ações, enquanto o ambiente é o mundo externo com o qual o agente interage. Em uma formulação discreta, a cada unidade de tempo t, o agente observa o estado atual do ambiente s_t e executa uma ação a_t, escolhida por meio de uma política π. Em seguida, o agente recebe do ambiente o estado seguinte s_{t+1} e a recompensa R_{t+1} obtida naquele instante, continuando o processo. O objetivo final do treinamento é encontrar uma política ótima π^* capaz de maximizar o retorno acumulado esperado de longo prazo, ponderado por um fator de desconto $\gamma \in [0, 1)$, que define o grau de preferência entre o acúmulo de recompensas a curto e longo prazo, sendo $\gamma = 0$ e $\gamma = 1$, respectivamente, os extremos que consideram apenas os ganhos imediatos e o acúmulo de todas as recompensas sem desconto para as futuras [Sutton and Barto 2018].

Dentre os diversos algoritmos que permitem descobrir uma política ótima, o Q-Learning é um dos mais utilizados e armazena os valores $Q(s, a)$ (que representa o retorno estimado de um par estado-ação) em uma tabela que é atualizada conforme o agente é treinado. Porém, em ambientes nos quais o espaço de estados possui alta dimensão (como o caso de imagens, jogos eletrônicos e vídeos), a demanda por memória para armazenar os estados na tabela Q torna-se inviável.

Diante desse cenário, o Deep Q-Network (DQN) surge como alternativa ao utilizar redes neurais profundas, como Redes Neurais Convolucionais (CNN), para transformar dados de entrada de alta dimensão (como imagens) em vetores de baixa dimensão e aproximar os valores da função $Q(s, a)$ com uma outra rede neural. Esta outra rede neural possui como entrada a representação de baixa dimensão do estado do ambiente e como saída o valor estimado da função $Q(s, a)$ para cada ação (o número de saídas da rede é igual ao número de ações). De forma a garantir a convergência do treinamento do modelo, implementa-se um *Replay Buffer*, que armazena transições de estado e recompensas passadas e as sorteia aleatoriamente em lotes, reduzindo a correlação temporal dos dados [Mnih et al. 2015].

2.2. Trabalhos Relacionados

A literatura mostra que pequenas perturbações visuais podem causar grandes impactos em problemas de visão computacional, sendo possível induzir modelos de classificação a erros alterando levemente a imagem [Szegedy et al. 2013] ou até mesmo mudando apenas um pixel [Su et al. 2019]. De modo semelhante, no contexto de aprendizado por reforço,

[Huang et al. 2017] mostra que ruídos adversariais imperceptíveis para um ser humano na imagem de entrada de um modelo de DQN podem levar o agente a escolher ações ruins. Ruídos adversariais são construídos de modo a induzir, intencionalmente, erros ou comportamento indesejado na saída do modelo.

Ruídos não adversariais, por outro lado, não são escolhidos especificamente com base em como prejudicar o modelo, podendo ser aleatórios, decorrentes de sensores, iluminação, oclusões naturais, dentre outros. Análises de robustez para este tipo de ruído também estão presentes na literatura. Em [Hendrycks and Dietterich 2019], é introduzido o dataset ImageNet-C, uma extensão do ImageNet com adição de corrupções visuais, como benchmark para análises de robustez. [Li et al. 2020] propõe uma arquitetura que decompõe as *feature maps* em componentes de baixa e alta frequência, descartando estes durante a inferência para aumentar a robustez ao ruído. Já em [Zhang et al. 2025], é utilizado um módulo de percepção de ruídos e um mecanismo de calibração para lidar com corrupções causadas por nuvens em imagens de sensoriamento remoto.

No contexto de jogos do Atari, [Farebrother et al. 2018] investiga a capacidade de generalização de agentes DQN a diferentes modos e dificuldades do mesmo jogo, observando quedas na recompensa quando os agentes são avaliados em configurações não vistas no treinamento. Entretanto, o trabalho não investiga especificamente a robustez a perturbações visuais.

Este trabalho concentra-se na análise da robustez de agentes DQN a perturbações visuais não adversariais no jogo Pong, avaliando a intensidade da degradação de desempenho para diferentes tipos e níveis de perturbações visuais. Experimentos adicionais em Boxing e Space Invaders também foram conduzidos para verificar se os efeitos observados também se manifestam em outros jogos.

3. Metodologia de Avaliação

O algoritmo Deep Q-Learning (DQN) é utilizado para treinar um agente no ambiente original, sem perturbação visual. Esse agente treinado tem seu desempenho de jogo avaliado quando submetido a jogos com diferentes perturbações visuais. A Figura 1 apresenta uma visão geral do fluxo experimental adotado no trabalho.

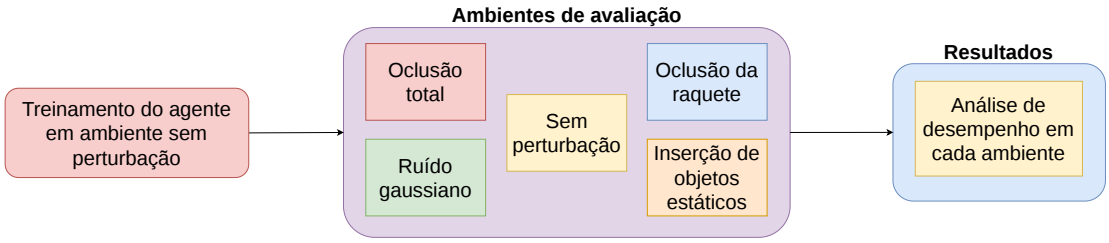


Figura 1. Fluxograma experimental do trabalho.

3.1. Bibliotecas utilizadas, arquitetura da rede neural e hiperparâmetros de treinamento

A criação do ambiente foi feita por meio da biblioteca Gymnasium [Towers et al. 2025], utilizando o ambiente `ALE/Pong-v5`, que incorpora os *sticky actions*, um mecanismo

que adiciona estocasticidade à dinâmica do jogo ao definir uma probabilidade de 0.25 de que a última ação seja executada novamente, em vez da nova ação selecionada pelo agente. A implementação do algoritmo Deep Q-Network (DQN) foi realizada por meio da biblioteca Stable-Baselines3 (SB3) [Raffin et al. 2021].

As observações do ambiente (telas do jogo), antes de serem fornecidas como entrada na rede neural, são passadas por uma etapa de pré-processamento. Os frames foram redimensionados para a resolução 84×84, convertidos para escalas de cinza, processados com *frame skip* igual a 4 (a sequência de frames observados pelo agente é construída a cada 4 frames de jogo). Além disso, foi utilizado o empilhamento de 4 frames consecutivos para fornecer informações temporais sobre o ambiente, como o deslocamento da bola e das raquetes. Dessa forma, a entrada da rede é composta por 4 frames consecutivos (considerando o frame skip) empilhados, resultando em uma observação de dimensão $(4 \times 84 \times 84)$. Um diagrama da etapa de pré-processamento é apresentado na Figura 2

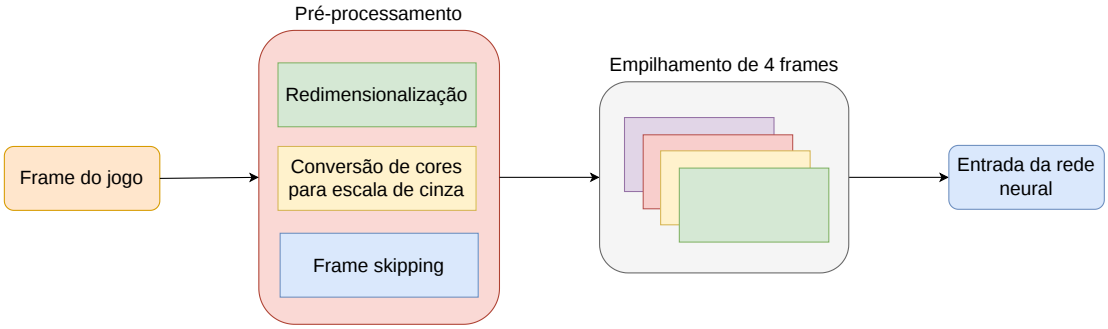


Figura 2. Fluxograma do pré-processamento da entrada da rede neural.

A política `CnnPolicy` do SB3 foi adotada para a DQN deste artigo. A arquitetura padrão da biblioteca é composta por 3 camadas convolucionais: a primeira camada possui 32 filtros com kernels de dimensão 8×8 e stride 4, seguida por uma segunda camada com 64 filtros, kernel 4×4 e stride 2, e uma terceira camada também com 64 filtros, kernel 3×3 e stride 1. Cada camada possui uma ativação ReLU, e a saída é achatada (flatten) em um vetor linear ao fim da última camada. Por fim, há uma camada totalmente conectada que recebe esse vetor e retorna um valor escalar para cada ação, correspondendo às estimativas de $Q(s, a)$ do DQN. No treinamento, foram utilizados os hiperparâmetros apresentados na Tabela 1.

3.2. Perturbações visuais

A fim de avaliar a robustez do agente perante perturbações visuais no ambiente, foram implementadas quatro perturbações distintas, que foram aplicadas aos frames (telas do jogo observadas pelo agente) antes de serem passadas como entrada para a rede neural que determina a ação a ser tomada naquele frame.

- O ruído gaussiano, apresentado na Figura 3, consiste em adicionar um valor aleatório em cada pixel da imagem de entrada da rede neural. Esse valor pertence a uma distribuição normal com média zero e diferentes desvios padrão. Para manter consistência na representação da imagem de entrada, os valores nos pixels modificados são limitados ao intervalo [0, 255].

Tabela 1. Hiperparâmetros de treinamento

Parâmetro	Valor
Learning rate	5×10^{-5}
Buffer size	$500\,000$
Learning starts	$50\,000$
Batch size	32
Gamma	0.99
Exploration fraction	0.05
Exploration final eps	0.01
Total timesteps	$20\,000\,000$

- A inserção de objetos estáticos, apresentada na Figura 4, consiste na adição de uma quantidade arbitrária de bolas visualmente semelhantes à bola do jogo original em lugares aleatórios do ambiente. As posições foram sorteadas uniformemente no início de cada episódio e permaneceram fixas durante todo o episódio de jogo. A região de inserção foi limitada para evitar as regiões do placar e as bordas da tela.
- Na oclusão da raquete adversária, ilustrada na Figura 5, foi feito um ocultamento da raquete adversária por meio da inserção de uma máscara que possui um valor de pixel igual ao fundo do ambiente. Dessa forma, o agente não conhece a posição da raquete do adversário.
- A oclusão total do lado adversário, apresentada na Figura 5, diferente da oclusão parcial, faz o ocultamento por meio da inserção de uma máscara de pixels pretos no lado do adversário. Repare que além do ocultamento da raquete do adversário, há o ocultamento em preto da região do lado do adversário.

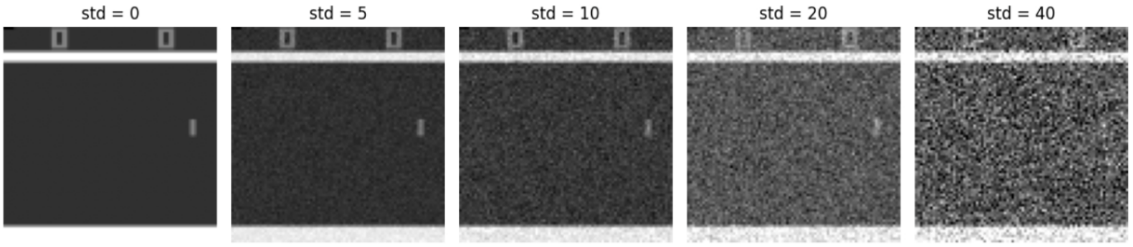


Figura 3. Exemplos de perturbação do ambiente com ruído gaussiano para diferentes valores de desvio padrão.

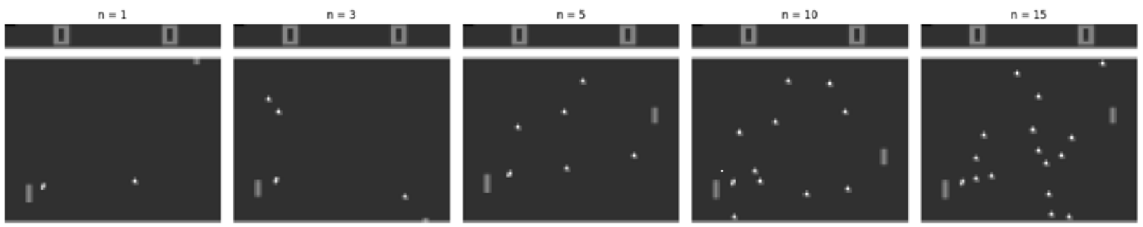


Figura 4. Exemplos de perturbação do ambiente com inserção de diferentes quantidades de bolas estáticas.

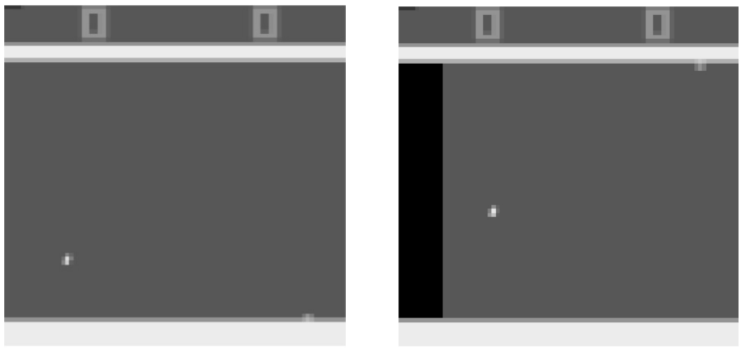


Figura 5. Exemplos de oclusão parcial, omitindo apenas a raquete adversária (esquerda), e oclusão total do lado do adversário (direita).

4. Resultados

Nessa seção, serão apresentados os resultados obtidos em dois treinamentos no jogo Pong: um no ambiente sem perturbação e outro com oclusão total do lado adversário. Também serão discutidos os treinamentos no ambiente sem perturbação em outros dois jogos: Boxing e Space Invaders. Os agentes foram treinados por 20 milhões de passos e com os mesmos hiperparâmetros. O desempenho foi medido por meio da média de episódios de avaliação.

4.1. Treinamento feito no ambiente original

Inicialmente, foi feito o treinamento do agente no ambiente sem perturbação. A Figura 9 apresenta a evolução da média de recompensa acumulada por episódio durante o treino em dois cenários: ambiente limpo e com oclusão. As curvas roxa e azul representam os desempenhos nos ambientes com e sem oclusão, respectivamente. Em ambos os casos, o desempenho foi medido por meio de avaliações periódicas, que foram realizadas a cada 500 mil passos, momento em que o agente executa ações determinísticas seguindo a política aprendida até aquele ponto.

Após o treinamento, o agente foi avaliado em 100 episódios no ambiente original, obtendo uma média de 19.50 ± 1.42 de recompensa. Posteriormente, foi feita a avaliação em ambientes com diferentes perturbações visuais, produzindo telas que nunca foram observadas antes pelo agente. Inicialmente, investigou-se os efeitos do ruído gaussiano. Foram adicionados ruídos com desvios padrão $\sigma \in \{1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 10, 20, 40\}$, além do caso sem ruído, $\sigma = 0$. Para cada σ, foram gerados 20 episódios de avaliação e calculados a média e desvio padrão das recompensas adquiridas. Na Figura 6, é possível observar que o desempenho do agente se mantém próximo de 19 até $\sigma = 2.5$. A partir de $\sigma = 3$, nota-se uma piora progressiva do desempenho e aumento do desvio padrão da recompensa média. Entre $\sigma = 4$ e $\sigma = 4.5$ houve uma queda abrupta da recompensa média, que passou de 3.50 para -11.50, indicando que a capacidade de o agente tomar decisões em tais ambientes foi prejudicada. Além disso, os desvios padrão das recompensas nesses dois casos são os que possuem maior valor numérico, mostrando instabilidade no comportamento do agente, já que em alguns episódios ele ganha e em outros, perde. Para valores de $\sigma \in \{10, 20, 40\}$ o desempenho é semelhante: recompensa média perto de -21 e desvio padrão da recompensa baixo, o que indica que o agente está perdendo com a pontuação mínima na maioria dos episódios de avaliação.

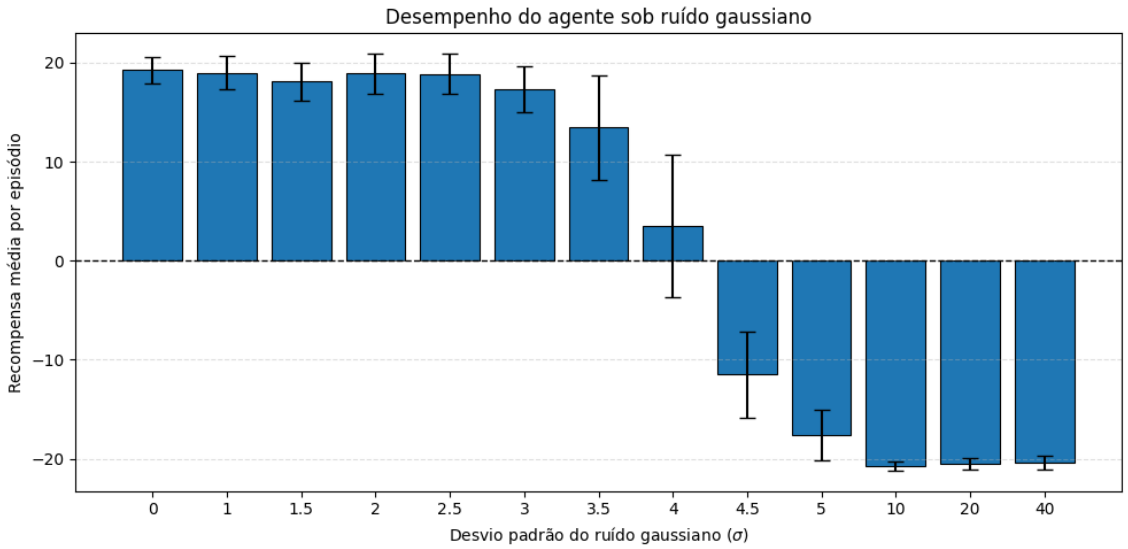


Figura 6. Recompensa média e desvio padrão do agente avaliado sob diferentes níveis de ruído gaussiano aplicado ao ambiente.

Para a investigação da adição de objetos estáticos em lugares aleatórios no ambiente, foram incluídas $n \in [0, 15]$ bolas idênticas à bola do jogo original. Após isso, foram gerados 20 episódios de avaliação para cada n, a partir dos quais foram calculados a média e desvio padrão das recompensas obtidas. De modo semelhante ao observado no experimento com ruído gaussiano, a recompensa média piorou de maneira progressiva conforme o número de objetos estáticos adicionados aumentou. Com $n \in \{1, 2, 3\}$ houve uma queda gradual da recompensa média acompanhada de um desvio padrão alto, indicando uma alta variabilidade do impacto das bolas falsas entre os episódios de avaliação. Uma possível explicação é que esse impacto dependa do posicionamento das bolas na tela, causando maior degradação quando dispostas em regiões mais críticas para a tomada de decisão do agente. A partir de $n = 4$, a recompensa média se aproxima de -21, indicando que o agente passa a falhar constantemente diante dessa perturbação visual. A Figura 7 ilustra o comportamento descrito.

Na análise dos efeitos da oclusão da raquete adversária e oclusão total do lado adversário, foram gerados 20 episódios para cada perturbação visual adicionada ao ambiente. O experimento com oclusão da raquete adversária teve como resultado uma recompensa média de 17.7, com desvio padrão igual a 2.81. Esse resultado sugere que o agente ainda é capaz de manter uma política eficiente, mesmo que a remoção da raquete tenha afetado um pouco o desempenho.

O experimento com oclusão total do lado adversário, por outro lado, levou a um impacto mais intenso no desempenho, tendo recompensa média de -20.95 e desvio padrão igual a 0.22. Esse resultado indica que o agente treinado no ambiente original utiliza informações visuais presentes no lado adversário para estimar valores Q associados aos pares estado-ação de modo adequado, e por isso, o agente teve um bom desempenho quando apenas a raquete do adversário foi coberta, pois essa configuração ainda preserva parte significativa da estrutura visual do ambiente original. No entanto, quando toda a região adversária é substituída por uma máscara de pixels pretos, a entrada fornecida à

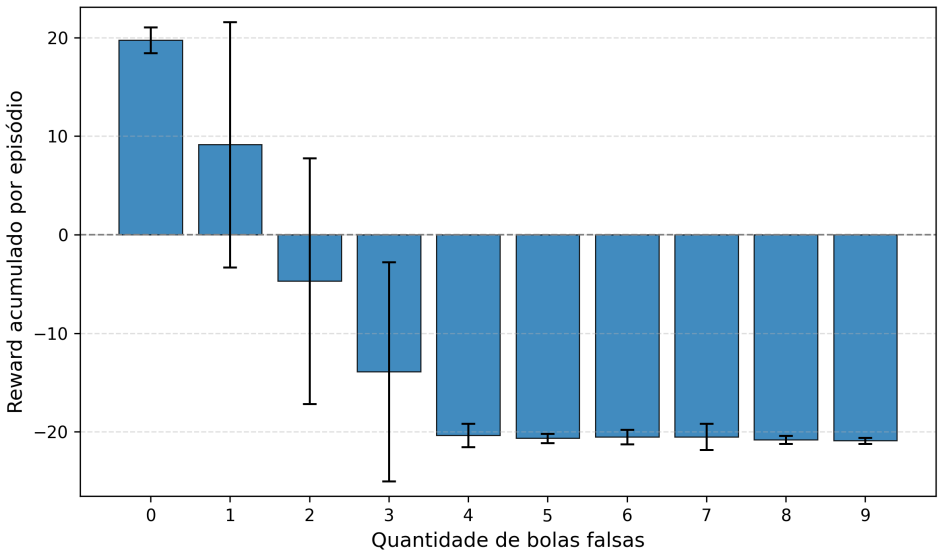


Figura 7. Recompensa média e desvio padrão do agente avaliado sob quantidades diferentes de réplicas estáticas da bola adicionadas ao ambiente.

rede neural difere de forma abrupta das observações vistas durante o treinamento, o que pode levar a estimativas imprecisas dos valores Q, impactando na escolha das ações do agente. A Tabela 2 mostra uma tabela comparativa desses resultados.

4.2. Treinamento em ambiente com perturbação visual

Foi realizado o treinamento de um segundo agente em um ambiente com oclusão total do lado adversário com o objetivo de investigar a capacidade de generalização deste agente ao ser avaliado em um ambiente sem perturbação. A Figura 8 descreve o fluxo deste experimento.

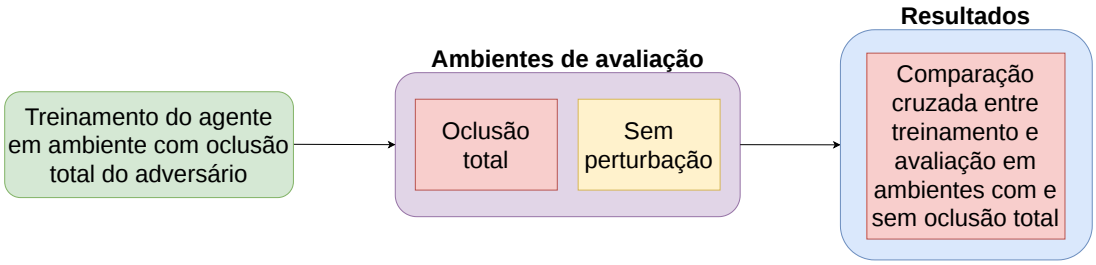


Figura 8. Fluxograma experimental para o agente treinado em ambiente com perturbação.

A Figura 9 apresenta a evolução da recompensa média acumulada por episódio nos treinamentos nos ambientes sem perturbação e com oclusão total do adversário. É possível observar que o treino foi mais instável no ambiente com oclusão do que no ambiente sem perturbação.

Após o treinamento, foi feita uma avaliação a partir de 100 episódios com o mesmo ambiente de treino (com oclusão total), no qual se obteve uma recompensa média de 17.86, com desvio padrão de 2.31. Além disso, o mesmo agente foi avaliado no ambiente sem oclusão por 20 episódios, obtendo uma recompensa média de -11.11, com

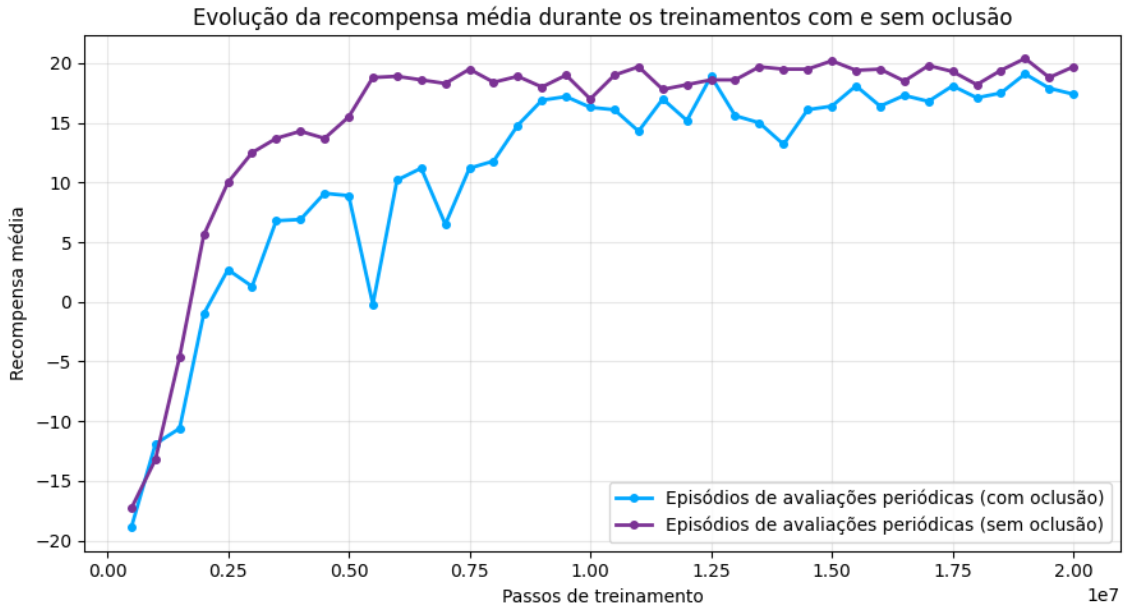


Figura 9. Evolução da recompensa média acumulada por episódio durante treino em ambiente sem e com perturbação.

desvio padrão de 3.15. Apesar de esse resultado representar uma queda significativa do desempenho em relação à avaliação no ambiente com perturbação, o declínio não foi tão acentuado quanto no caso inverso, com o agente treinado no ambiente original e avaliado com oclusão total. Isso sugere que o treinamento com oclusão total fez o modelo depender menos das informações visuais do lado adversário da tela. Dessa forma, quando o agente foi avaliado no ambiente sem perturbações, a adição das informações do lado adversário da tela alterou as imagens de entrada da rede neural, mas afetou com menos intensidade as suas saídas, comparado ao caso inverso.

Avaliação \ Treino	**Sem ruído**	**Oclusão total**
Sem ruído	19.50 ± 1.42	-11.11 ± 3.15
Oclusão total	-20.95 ± 0.22	17.86 ± 2.31
Oclusão parcial	17.70 ± 2.81	—

Tabela 2. Recompensa média sob diferentes condições de treino e avaliação.

4.3. Treinamento e avaliação em outros jogos

Nessa seção, estenderemos a discussão anterior para dois outros jogos: Space Invaders e Boxing. Em ambos os casos, foi realizado o treinamento em ambiente limpo e avaliação em ambientes com diferentes níveis de perturbação visual gaussiana, com $\sigma \in \{1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 10, 20, 40\}$, conforme ilustrado pela Figura 10. De forma geral, o comportamento dos agentes foi similar ao fenômeno que aconteceu no Pong: a recompensa média diminuiu conforme o ruído aumenta. No Boxing, o desempenho permaneceu estável para $\sigma < 10$, com recompensas próximas à condição sem ruído. A partir desse ponto, a degradação se torna mais evidente, atingindo uma recompensa média de 6.10 com $\sigma = 40$. Já em Space Invaders, a redução de desempenho foi de forma

mais gradual, decaindo de 522.85 para $\sigma = 0$ para 396.65, 329.10, 247.15 e 182.25 em $\sigma = 5$, 10, 20 e 40, respectivamente.

Nesse sentido, os resultados indicam que a queda de desempenho do agente perante a ruídos não é exclusiva do Pong, mas o impacto exato causado por cada nível de ruído visual depende do ambiente. Enquanto o Pong apresenta uma queda abrupta com níveis de ruído mais baixos, o Boxing registra maiores degradações apenas em níveis mais altos de ruído, mantendo seu desempenho próximo do original para valores mais amplos de σ. Em Space Invaders, o comportamento é intermediário, definido pela diminuição progressiva da recompensa média à medida que a intensidade do ruído aumenta.

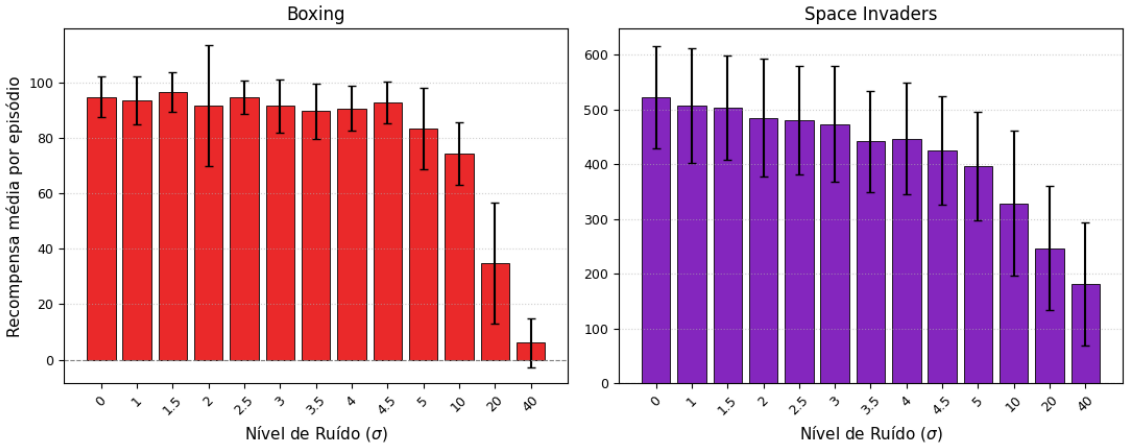


Figura 10. Recompensa média e desvio padrão do agente avaliado sob diferentes níveis de ruído gaussiano aplicado ao ambiente para os jogos Boxing (à esquerda) e Space Invaders (à direita).

5. Conclusão

Este trabalho avaliou a robustez de uma Deep Q-Network (DQN) em cenários com diferentes perturbações visuais, utilizando como ambiente de treinamento e avaliação o jogo eletrônico Pong, do Atari. Os resultados demonstram as limitações fundamentais de redes neurais convolucionais tradicionais ao serem testadas e avaliadas em ambientes com distribuição dos dados de entrada diferente dos ambientes nos quais foram treinadas.

A degradação severa da recompensa média obtida pelo agente revela que o modelo mapeia o conjunto ótimo de ações através do vetor de entrada da rede neural. Por isso, perturbações como ruído gaussiano e oclusão parcial de tela, que seriam facilmente ignorados por operadores humanos, traduzem-se em alta perda de desempenho para o agente. Não há aprendizagem relacional: o modelo processa apenas os pixels da imagem, sem compreender conceitos por trás de entidades como a bola e as raquetes.

Além disso, é interessante reparar que, ao treinar os modelos com a oclusão total do lado adversário, ele foi capaz de obter uma recompensa média alta quando avaliado em um ambiente com este mesmo tipo de ruído. Outro ponto de destaque é que o desempenho desse agente treinado em um ambiente com perturbação visual e avaliado em um ambiente sem ruído não é idêntico ao da situação contrária.

Ademais, ao fazer experimentos em outros ambientes de Atari, foi possível observar que a queda de desempenho sob ruído visual ocorreu de formas diferentes: de forma

abrupta em determinados jogos e progressivo em outros. Esse comportamento pode ser explicado pela variação na quantidade de informação visual necessária em cada ambiente para selecionar ações adequadas, tornando alguns agentes mais tolerantes a perturbações da representação de entrada.

Como direcionamento para trabalhos futuros, recomenda-se a investigação e implementação de técnicas de robustez de ruídos no contexto de aprendizado por reforço e o desenvolvimento de agentes robustos a perturbações visuais não observadas durante o treinamento.

Referências

Farebrother, J., Machado, M. C., and Bowling, M. (2018). Generalization and regularization in dqn. *CoRR*, abs/1810.00123.

Hendrycks, D. and Dietterich, T. G. (2019). Benchmarking neural network robustness to common corruptions and perturbations. *CoRR*, abs/1903.12261.

Huang, S., Papernot, N., Goodfellow, I. J., Duan, Y., and Abbeel, P. (2017). Adversarial attacks on neural network policies. In *International Conference on Learning Representations (ICLR) Workshop Track*.

Li, Q., Shen, L., Guo, S., and Lai, Z. (2020). Wavelet integrated cnns for noise-robust image classification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7243–7252.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533.

Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., and Dormann, N. (2021). Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8.

Su, J., Vargas, D. V., and Sakurai, K. (2019). One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. The MIT Press, second edition.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Towers, M., Kwiatkowski, A., Balis, J., De Cola, G., Deleu, T., Goulão, M., Andreas, K., Krimmel, M., KG, A., Perez-Vicente, R., Terry, J., Pierré, A., Schulhoff, S., Tai, J. J., Tan, H., and Younis, O. G. (2025). Gymnasium: A standard interface for reinforcement learning environments. In *Advances in Neural Information Processing Systems*, volume 38.

Zhang, P., Cheng, G., Lang, C., Xie, X., and Han, J. (2025). Nirnet: Noise incentive robust network in remote sensing object detection under cloud corruption. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–13.