

UTILIZAÇÃO DE CLASSIFICAÇÃO DE IMAGENS EM TINYML PARA AUXÍLIO NA ALIMENTAÇÃO DE AVES

Luísa Gaino Faria¹ (EG), Mariana Natalie Tonini de Melo² (EG), Egon Luiz Muller (PQ)¹

¹Universidade Federal de Itajubá - Campus Itajubá.

Palavras-chave: Aves, Classificação de imagens, Deep Learning, Redes convolucionais, TinyML

Introdução

A alimentação de determinadas espécies de aves ao ar livre pode ser uma tarefa que não possui muito controle. Não há como garantir que a comida que foi destinada a uma determinada espécie atinja de fato o destinatário correto. Isso porque, ao ar livre, diversas outras espécies podem entrar em contato com esse alimento e usurpá-lo da espécie para a qual era pretendido.

Nesse projeto, analisou-se a alimentação de patos em um lago: a comida destinada aos patos era frequentemente roubada por pombos, gatos e cachorros e, por isso, tornava-se necessário disponibilizar uma quantidade muito maior de comida para os patos para que eles fossem propriamente alimentados. Essa situação gerava, portanto, um desperdício muito grande de comida, já que a comida não atingia de fato a espécie pretendida.

Nos últimos anos, têm-se visto um grande aumento nas aplicações que utilizam Tiny Machine Learning (TinyML) para resolver problemas domésticos, facilitando e automatizando tarefas. A adoção de soluções em TinyML permite a construção de aplicações com baixo consumo de energia, baixo custo e que executam as tarefas localmente, não dependendo, necessariamente, da internet para operar.

Por isso, achou-se adequado usar uma solução que utiliza esta tecnologia para projetar um alimentador automático que utiliza classificação de imagens, por meio de um modelo de Deep Learning com redes convolucionais, que objetiva detectar qual é a espécie que está de fato desfrutando dos alimentos: o modelo, que será embarcado em um alimentador, detecta a espécie que se aproxima e só permite que a comida seja liberada caso condiz com a espécie esperada.

O uso de TinyML nessa aplicação permite que o dispositivo possa operar localmente, já que permite o processamento dos dados na borda, além de possibilitar a utilização de baterias, por não possuir consumo de energia elevado. A utilização dessa tecnologia também possibilitou que o modelo fosse embarcado em uma

placa de baixo custo, já que não é uma aplicação que exige grande poder de processamento.

Até 2012, técnicas mais clássicas de Machine Learning eram consideradas o padrão, mas na competição anual ImageNet, que usa mais de 14 milhões de imagens a serem classificadas e mostra os melhores modelos da competição, um modelo se destacou muito mais do que seus concorrentes, a AlexNet de Krizhevsky et al. (2012), é um modelo de Deep Learning que usa uma rede convolucional com 8 camadas, obtendo um erro 10,8% menor que o segundo colocado na competição, atraindo todos os olhares para redes convolucionais para processamento de imagens.

Em 2014, a rede Inception de Szegedy et al. (2015) surgiu, com o foco de reduzir os custos computacionais usando múltiplas camadas de convolução a paralelização de operações.

As redes acima foram pensadas em aplicações para *hardwares* potentes e alimentados à fontes. Em 2017, o time do Google desenvolveu a rede MobileNet, por Howard et al. (2017), para maximizar a performance em dispositivos com processamento e memória limitados. Essa arquitetura usa um tipo de convolução chamado de *Depthwise Separable Convolution*, que constitui-se de uma convolução do tipo *depthwise convolution* com *Batch Normalization* e uma função de ativação *ReLU*, seguida por uma convolução do tipo *pointwise convolution* com *Batch Normalization* e uma função de ativação *ReLU*, reduzindo os custos computacionais e o número de parâmetros da rede.

Existem uma série de pesquisas que construíram aplicações similares com o uso de TinyML, no entanto, para outros propósitos de utilização.

Sabella et al. (2022) produziram um sistema de classificação acústica utilizando TinyML, envolvendo o monitoramento de animais, principalmente as espécies ameaçadas de extinção, em seu habitat natural. Esse sistema consegue identificar a densidade populacional da fauna selvagem, garantindo o monitoramento das espécies sem a necessidade da presença humana nesses ecossistemas.

Diab and Rodriguez-Villegas (2022) também

utilizou modelos de classificação de imagens, nesse caso, para uso médico. Esse trabalho utilizou da plataforma Edge Impulse para o treinamento do modelo que foi, também, embarcado e tinha como objetivo a utilização do classificador de imagens como ferramenta de diagnóstico médico.

Metodologia

Para a criação do classificador de imagens, foi utilizado a plataforma Edge Impulse, que funciona para treinar, converter e quantizar modelos de Deep Learning baseados no framework Tensorflow, gerando uma biblioteca na linguagem C para utilização em microcontroladores. Para a aquisição de dados, foram utilizados dois datasets disponíveis na plataforma Kaggle, o bird_species e o Animal Detection Images Dataset, sendo o primeiro utilizado com o intuito de diferenciar as aves entre si, enquanto o segundo dataset, apesar de também auxiliar na diferenciação entre diferentes espécies de aves, também teve o intuito de diferenciar as aves das espécies em outras classes.

Nem todas as imagens do dataset foram utilizadas para treinar o modelo inicial, já que o intuito do projeto é apenas a detecção de patos. Para o treinamento do modelo, as imagens foram redimensionadas para 96x96, essas imagens foram então submetidas para o treinamento utilizando Transfer Learning, que utiliza modelos já treinados de Machine Learning para criar novos modelos com dados diferentes.

Após o modelo ter sido treinado, foram realizados testes na própria plataforma Edge Impulse. Após esses testes, verificando a precisão do modelo, foi então gerada uma biblioteca na linguagem C, para ser embarcado. O modelo foi embarcado em uma placa ESP32-Wrover TTGO T-Camera, que é equipada com uma câmera, um sensor de aproximação e um display LCD, ilustrada na Figura 1.

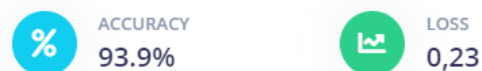


Figura 1 – Placa utilizada para embarcar o modelo.

Resultados e discussão

O modelo, após o treinamento, resultou em uma precisão de 93,9%, apresentando a matriz de confusão da Figura 2 e o gráfico da dispersão dos dados.

Last training performance (validation set)



Confusion matrix (validation set)

	DUCK	NOT-DUCK
DUCK	84.0%	16.0%
NOT-DUCK	2.9%	97.1%
F1 SCORE	0.87	0.96

Figura 2 – Matriz de confusão resultante do treinamento.

Depois, ao inserir os dados de teste, obteve-se uma precisão de 89,98%, apresentando uma pequena queda de precisão, mas que acredita-se que não será de grande significância no modelo embarcado.

Além disso, o tempo de inferência foi de 95ms conforme vistos na Figura 3, dessa forma garantindo que a alimentação dos patos ocorreria sem atrasos, sem correr riscos de que o pato possa sair de perto do alimentador sem que tenha se alimentado. Ainda na Figura 3, vemos que a memória ocupada pelo modelo não é muito grande, ocupando 12,5% da memória flash da ESP-32, o que garante que o modelo poderá operar sem problemas de falta de memória, que poderia sobrecarregar o sistema.

On-device performance



Figura 3 – Dados dos testes.

Ao embarcar o modelo na placa, havia a preocupação de que, devido a baixa resolução das imagens capturadas pela câmera, se tornasse mais difícil de detectar os animais do que quando comparado aos testes feitos no Edge Impulse. No entanto, o modelo embarcado apresentou precisão parecida com a vista na plataforma, dado que foi possível reconhecer as aves em 90% dos testes realizados com a placa.

Conclusões

De acordo com o visto na seção de resultados, o modelo pode ser embarcado cumprindo com o proposto, reconhecendo os patos com precisão similar ao esperado. Isso mostra que essa é uma aplicação que, de fato, pode ser utilizada para facilitar a alimentação das aves, podendo reduzir o desperdício nesse processo. O custo do protótipo é baixo, possibilitando que a aplicação seja mais acessível ao público em geral.

Agradecimentos

Gostaríamos de expressar nossos agradecimentos à Universidade Federal de Itajubá e ao PET-TEC, por proporcionar o ambiente acadêmico e os recursos necessários que possibilitaram a realização deste projeto. Além disso, estendemos nossa gratidão ao FNDE, por todo apoio.

Referências

- Diab, M.S. and Rodriguez-Villegas, E. (2022). Performance evaluation of embedded image classification models using edge impulse for application on medical images. In 2022 44th Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC), 2639–2642. doi:10.1109/EMBC48229.2022.9871108.
- Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.
- Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc. URL <https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>.
- Sabbella, H.R., Nair, A.R., Gumme, V., Yadav, S.S., Chakrabartty, S., and Thakur, C.S. (2022). An always-on tinyml acoustic classifier for ecological applications. In 2022 IEEE International Symposium on Circuits and Systems (ISCAS), 2393–2396. doi:10.1109/ISCAS48785.2022.9937827.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1–9. doi:10.1109/CVPR.2015.7298594.