

# ESTIMATIVA DA ALTURA DE VOO DE UM VETOR AÉREO POR MEIO DE RECONHECIMENTO DE PADRÕES E INTELIGÊNCIA ARTIFICIAL

Gabriel Dietzsch<sup>1</sup>, Elcio Hideiti Shiguemori<sup>2</sup>, Fabiano da Cruz Nogueira<sup>3</sup> e Rafael Marinho de Andrade<sup>4</sup>

<sup>1234</sup>IEAv-Instituto de Estudos Avançados, Trevo Coronel Aviador José Alberto Albano do Amarante, nº 1 – Putim, São José dos Campos-SP e <sup>1</sup>dietzschgd@fab.mil.br; <sup>2</sup>elcioehs@fab.mil.br; <sup>3</sup>nogueirafcn@fab.mil.br; <sup>4</sup>rafael.andrade23@fatec.sp.gov.br

## RESUMO

Este trabalho explora o uso da Inteligência Artificial e Reconhecimento de Padrões para compor o processo de estimativa da altura de voo de um vetor aéreo (drone, avião, helicóptero, entre outros) em um processo de navegação autônoma. Verifica-se neste trabalho que, desde que os parâmetros da câmera sejam conhecidos, a rede neural convolucional empregada seja satisfatoriamente treinada e o reconhecimento de padrões de alvos de interesse com suas métricas sejam determinados, a estimativa da altura de voo resultante pode ser empregada como método de navegação autônoma nos vetores aéreos e, assim, compor mais uma alternativa para mitigar a negação ou interferência nos sinais GNSS. A rede neural convolucional treinada identificou 90% dos alvos de interesse e a estimativa da altura de voo apresentou resultados satisfatórios.

**Palavras-chave** – Inteligência Artificial, YOLO, Altura de voo, Vetor Aéreo, Navegação Autônoma.

## ABSTRACT

*This work explores the use of Artificial Intelligence and Pattern Recognition to compose the process of estimating the flight height of an aerial vector (drone, plane, helicopter...) in an autonomous navigation process. It is verified in this work that, as long as the camera parameters are known, the neural network used is satisfactorily trained and the pattern recognition of targets of interest with their metrics are determined, the resulting flight height estimation can be used as a method of autonomous navigation in the aerial vectors and, thus, compose one more alternative to mitigate the denial or interference in the GNSS signals. The trained neural network identified 90% of the targets of interest and the flight height estimation showed satisfactory results.*

**Key words** – Artificial Intelligence, YOLO, Height of flight, Aerial Vector, Autonomous Navigation.

## 1. INTRODUÇÃO

A Inteligência Artificial (IA) pode ser considerada como a inteligência demonstrada por máquinas, em oposição à inteligência natural exibida por animais, incluindo humanos. A pesquisa de IA foi definida como o campo de estudo de agentes inteligentes, que se refere a qualquer sistema que percebe seu ambiente e realiza ações que maximizam sua chance de atingir seus objetivos [1]. Apresentamos neste trabalho que o uso da IA, em especial as técnicas

de aprendizado de máquina, ocupa um espaço determinante para o sucesso do reconhecimento de padrões utilizado na navegação autônoma de um vetor (veículo) aéreo. A sua aplicação para a determinação de um dos principais fatores da navegação, o conhecimento da altura de voo, pode ser traduzida como a elevação da capacidade humana em estimar a altura e, ao mesmo tempo, uma redenção sobre a dependência tecnológica dos Sistemas Globais de Navegação por Satélites (GNSS).

Para que um veículo aéreo cumpra os principais requisitos de segurança para a sua navegação, além de estimar onde ele está, conhecer sua localização em termos de planimetria, ele também necessita conhecer a sua altura em relação ao solo por motivos: evitar colisão, ou queda. Altura, que é diferente de Altitude, consiste na distância vertical de um nível, ponto ou objeto considerado como ponto e uma determinada referência [2]. Tanto a planimetria quanto a altimetria são facilmente obtidas por meio dos GNSS. Contudo, o sistema possui vulnerabilidades. Segundo [3], as vulnerabilidades dos sinais dos GNSS foram evidenciadas 2001, quando usuários do GPS civil foram alertados que o sistema havia se mostrado vulnerável à interferência de transmissões de baixa frequência mais potentes, quer seja por meio de bloqueio, da imitação ou da retransmissão maliciosa do sinal. A partir de então, vários relatos de interferência intencional (criminoso) ou acidental aumentaram significativamente. No Brasil, onde existem apenas projetos de lei sobre o assunto, estão disponíveis diversos bloqueadores (*jammers*) de GPS à venda. Nos Estados Unidos da América, embora não existam estatísticas disponíveis, estima-se que existam milhares desses dispositivos de baixo custo (menos de US\$150) em uso nas suas estradas, apesar de sua venda e uso serem ilegais. Uma maneira de contornar esse problema é utilizar imagens aéreas obtidas por esses veículos e realizar um processamento em tempo real para estimar a posição no espaço por meio do uso intenso de Redes Neurais.

Então, o objetivo deste trabalho é realizar a identificação de padrões para estimar exclusivamente a altura de voo de um veículo aéreo por meio de técnica de aprendizagem de máquina, com uso de redes neurais convolucionais, com vistas ao apoio à navegação aérea autônoma. A metodologia, que será apresentada a seguir, utiliza imagens nadirais obtidas por Drone e conhecimento humano para “ensinar” o que a máquina deve identificar.

## 2. MATERIAL E MÉTODOS

Nesta seção serão apresentados os materiais e métodos utilizados para cumprir o objetivo proposto. Espera-se, ao final: identificar, com maior exatidão possível, duas faixas

consecutivas do conjunto de faixas de pedestres; encontrar parâmetros de saída adequados para estabelecer relação Mundo Real Vs Imagem; e realizar um cálculo para estimar a altura de voo.

## 2.1. YOLO

O uso de técnicas de Aprendizagem de Máquina para este trabalho se deve ao fato de que elas podem ser utilizadas em “tempo real”, “compreendem” cenários distintos e identificam com certo grau de confiabilidade o alvo desejado. Atualmente, o *YOLO (YouOnlyLookOnce)* é considerado o estado da arte em detecção de objetos em tempo real. É o detector de objetos com maior acurácia de acordo com os testes realizados utilizando o dataset MS COCO [4]. O *YOLO* utiliza uma rede neural convolucional para o processamento das imagens e extração da informação que se deseja, não é um classificador tradicional que foi reajustado para ser um detector de objetos, mas sim foi feito para ser capaz de rodar por toda a imagem apenas de uma vez, fazendo isso de uma forma inteligente. Outro fator relevante para a escolha do *YOLO* para a resolução do problema apresentado é que este método de treinamento de rede permite um treinamento multi-escala e diferentes resoluções. Embora a velocidade de treinamento seja mais lenta quando o tamanho da entrada for grande, a velocidade de treinamento é mais rápida quando o tamanho da entrada for pequeno. O treinamento pode melhorar a precisão, portanto, busca-se um bom equilíbrio entre precisão e velocidade [5].

A versão do *YOLO* utilizada neste trabalho foi a V4. Em [6] é possível encontrar todo detalhamento dessa rede neural e, inclusive a comparação desta frente às outras versões. O objetivo do *YOLO V4* neste trabalho é identificar faixas de pedestres. Por quê? Porque são referências dentro de cenários urbanos e possuem um padrão fixo para realização de suas pinturas. Suas larguras, comprimento e distância entre duas consecutivas são parametrizadas e variam dentro de padrões conhecidos. Esses padrões podem ser explorados no Manual Brasileiro de Sinalização de Trânsito [7].

O treinamento da rede *YOLO* foi realizado em ambiente do *Google Colab*. Inicialmente, o tratamento foi realizado com as fotos em resolução 4k, isso demandou tempo de processamento e a possibilidade de o ambiente ser desconectado era grande. Decidiu-se, então, redimensionar as imagens para o tamanho  $416 \times 277$  pixels. Esse redimensionamento fez o tempo inicial de processamento cair de 28h para 21h. Cabe ressaltar que havia o receio de se perder o trabalho de *labeling* realizado em etapa anterior mas os *boundingboxes* são relativos à imagem original e permanecem no local correto desde que a proporção da imagem seja mantida.

## 2.2. Tomada das imagens para o treinamento

Para gerar o *dataset* do treinamento da rede foi utilizado o Drone Phantom 4, imagens em resolução 4k, com alturas de voo variadas. Também foi tomado o cuidado de variar a rotação das tomadas das imagens. Assim, o *dataset* possui boa variedade de escalas e rotações para o mesmo tipo de alvo (faixas de pedestre) e, assim, simultaneamente repassar para

a rede condições muito próximas às aquisições de imagens durante um voo real.

Todo *dataset* criado para o treinamento da rede foi pautado nas características que as imagens aéreas carregam. Essas características foram satisfatoriamente elencadas por [8], a saber: Diversidade de escala: a altura de disparo dos *UAVs* varia de dezenas de metros a quilômetros, resultando em uma ampla faixa de tamanhos de objeto semelhante no solo; Especificidade da perspectiva: as perspectivas das imagens aéreas são basicamente vistas de alta altitude, enquanto a maioria dos conjuntos de dados convencionais são perspectivas ao nível do solo; Objeto pequeno: os objetos de imagens aéreas geralmente são apenas algumas dezenas ou mesmo alguns *pixels*, então sua quantidade de informação também é menor; Multidirecional: as imagens aéreas são tiradas da visão de um pássaro e a direção dos objetos é incerta (enquanto a direção do objeto no conjunto de dados convencional tende a ter certeza, como os pedestres geralmente estão na vertical); e Alta complexidade de fundo: as imagens aéreas possuem um grande campo de visão e podem conter uma variedade de fundos, o que terá uma forte interferência na detecção de objetos.

Pelas razões mencionada, muitas vezes é difícil treinar um classificador ideal em conjuntos de dados convencionais para as tarefas de detecção de objetos em imagens aéreas. Portanto, é necessário um conjunto de dados especializado. A leitura deste trabalho revelará que o conjunto de dados de imagens aéreas adquirido foi adequado para treinamento do *YOLO V4*. Foram tomadas 300 imagens e essas foram divididas da seguinte maneira: 240 para o treinamento da rede e 60 para a validação.

## 2.3. Processo de rotulação dos alvos na imagem

Após a tomada das imagens, procedeu-se para o processo de rotulação dos alvos que, de forma simples, é a etapa onde mostramos para a máquina o que queremos que a rede identifique ao final do treinamento. Para este caso, foram desenhados aproximadamente 2.500 *boundingboxes* (contorno que destaca um objeto em uma imagem), ou caixas rotuladas, contendo duas faixas consecutivas da faixa de pedestres fotografada. Ao final do treinamento é esperado que a rede consiga identificar *boundingboxes* similares a essas.

Cada *boundingbox* na imagem consiste nos seguintes atributos: Largura (*w*), Altura (*h*), Classe (por exemplo, pessoa, carro, semáforo, faixa de pedestre etc.) e Centro da caixa delimitadora (*bx,by*). O *YOLO* usa uma única regressão de caixa delimitadora para prever a altura, largura, centro e classe de objetos. Para isso, entra em cena a *IntersectionofUnion (IOU)* que descreve como as caixas se sobrepõem. O *YOLO* usa a *IOU* para fornecer uma caixa de saída que envolve os objetos perfeitamente. Cada célula da grade é responsável por prever as caixas delimitadoras e suas pontuações de confiança. O *IOU* é igual a 1 se a caixa delimitadora prevista for igual à caixa real. Esse mecanismo elimina caixas delimitadoras que não são iguais à caixa real.

### 2.4. Método de estimativa da altura de voo

Para se atingir o objetivo do cálculo da altura de voo, faz-se mister o conhecimento de alguns parâmetros da câmera utilizada. Para este trabalho foi necessário saber tamanho da imagem (5.472 × 3.648 pixels) e o FOV (*FieldofView*) da câmera, 84, de acordo com o [9]. Para o cálculo da estimativa da altura de voo (*h*) é utilizada equação (1):

$$h = D / \tan(B * P) \tag{1}$$

Onde:

**D** = tamanho do alvo no solo, 1m (conhecido de antemão e retirado de [7]);

**B** = 0,0127 = ângulo visual de 1(um) pixel. Obtido da relação FOV / Diagonal da Imagem em pixel; e

**P** = Tamanho em pixels do alvo na imagem (que será obtido por meio do *boundingbox* identificado pelo YOLO.

## 3. RESULTADOS

### 3.1. Resultado do treinamento do YOLO V4

O treinamento da rede rodou por cerca de 5.200 épocas e atingiu 90% de acurácia nos melhores pesos ajustados. O gráfico composto pelos dados do treinamento e dos resultados pode ser observado na Figura1. Em vermelho está representada a acurácia do treinamento ao longo das épocas e, em azul, o *loss*, que mostra o quão mal o modelo está indo no treinamento, quanto menor for o *loss*, melhor é o funcionamento do modelo.

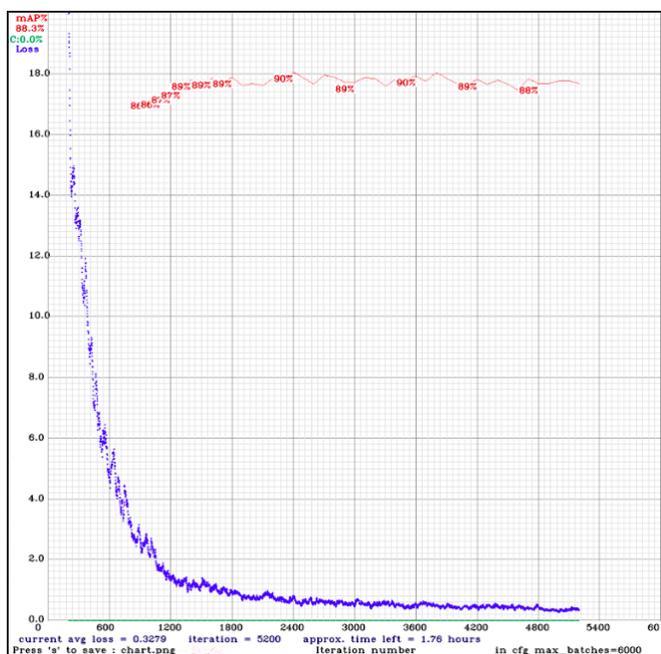


Figura 1: Gráfico do treinamento do YOLO

### 3.2. Resultado da identificação das faixas

Após o sucesso no treinamento da rede, passou-se para a etapa de ativação, ou seja, etapa onde uma imagem qualquer é apresentada para a rede e ela deve apresentar uma imagem

com um *boundingbox* evidenciando o alvo (duas faixas consecutivas da faixa de pedestres).

Esta etapa de identificação do alvo também deve fornecer subsídios para o cálculo da altura. Esses subsídios advém do tamanho do *boundingbox* (largura e comprimento) e representam o tamanho em pixel da faixa que será utilizado na próxima etapa do trabalho. Na Figura 2 está um exemplo de saída da ativação da rede neural. Para o cálculo da altura de voo sempre o menor valor dessas duas medidas será usado pois esse corresponde ao comprimento com maior garantia de não ter sofrido alteração em relação ao [7].

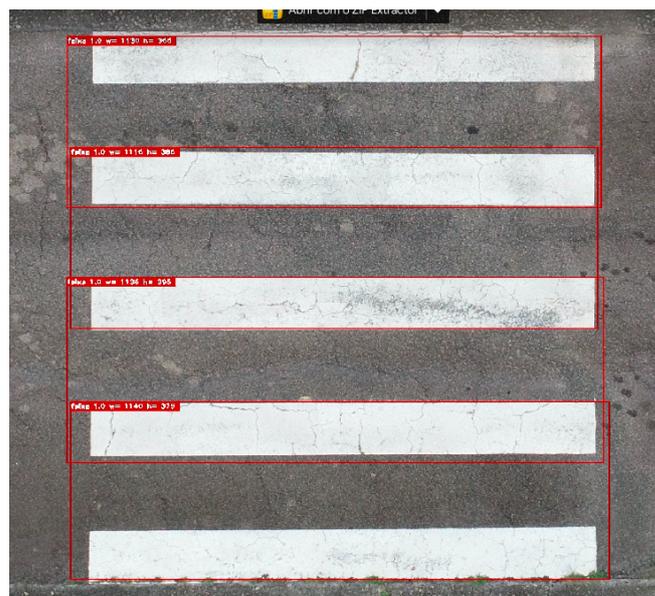


Figura 2: Exemplo de uma ativação da rede

### 3.3. Resultado da estimativa das alturas de voo

A Tabela 1 apresenta os resultados das estimativas de voo. As colunas são descritas na sequencia.

Foto	Altitude de voo	Delta h	Dist Média Faixas (pixels)	Altura de voo calculada	Delta h'	Erro em Delta (h - h')
1	695	0,00	957,00	4,62	0,00	0,00
2	696	1,00	720,00	6,18	1,56	-0,56
3	697	1,00	865,00	5,12	-1,06	2,06
4	697	0,00	833,00	5,32	0,20	-0,20
5	703	6,00	390,50	11,46	6,14	-0,14
6	704	1,00	348,33	12,85	1,39	-0,39
7	737	33,00	98,00	45,77	32,91	0,09
8	740	3,00	176,50	25,40	-20,36	23,36
9	758	18,00	121,50	36,91	11,51	6,49
10	760	2,00	77,50	57,88	20,96	-18,96

Tabela 1: Estimativa da altura de voo

Na primeira coluna apresenta-se o identificador da imagem; Na segunda coluna consta a altitude de voo em metros; A coluna "Delta h" é a diferença de altitude da linha atual menos a linha anterior em metros; A coluna "Dist média faixas", expressa em pixels, corresponde à média das menores dimensões encontradas para a imagem na ativação da rede; A coluna "altura de voo calculada" foi obtida por meio da equação 1; A coluna "Delta h'" é similar à coluna "Delta h" e calculada da mesma forma. Espera-se que essas colunas

apresentem valores iguais; Essa diferença entre os valores de Delta  $h'$  e Delta  $h$  fica evidenciada na coluna "Erro de Delta".

#### 4. DISCUSSÃO

Com relação ao resultado do treinamento do *YOLO V4*, nota-se que o *loss* do modelo cai rapidamente. Esse fenômeno pode ser atribuído ao fato de que estamos tratando da identificação de uma figura rudimentar. Nota-se também que a precisão não demora a se estabilizar. Podendo até mesmo o treinamento ser interrompido com aproximadamente 3000 épocas.

A identificação das faixas obteve êxito e foi possível identificar os alvos até mesmo em fotos oblíquas e de diferentes fontes. Contudo, essas imagem não podem ser utilizadas para a estimativa da altura de voo pelo fato de que a equação (1) está pautada em visada nadiral, ou seja, diretamente para baixo e na vertical. Foi constatado que o tamanho das *boundingboxes* podem variar e abranger mais *pixels* que o necessário, apresentando métricas além das faixas; ou menos que o necessário, neste caso cortando um pouco dos *pixels* que seriam relevantes para o cálculo da altura. Sobre a correta identificação e desenho da janela de identificação, constata-se que a forma ideal para a ativação da rede seria imagens com as faixas exatamente na vertical ou exatamente na horizontal. Isso proporciona melhor acurácia do tamanho do comprimento e da largura das faixas.

Sobre o resultado das estimativas das alturas de voo, nota-se na coluna "Dist média faixas", da Tabela 1, que os valores diminuem à medida em que a altitude aumenta, demonstrando coerência até a foto 8; na coluna "altura de voo calculada", bem como a coluna anterior, apresenta coerência até a imagem 8, ocorrendo um erro na imagem 3. Faz-se a comparação entre as diferenças das altitudes e das alturas, que devem ser praticamente iguais caso a metodologia se mostre capaz de determinar a altura de voo; a coluna "Erro de Delta" também apresentou coerência até a imagem 8.

#### 5. CONCLUSÕES

Diante do exposto, resta para a conclusão deste trabalho os seguintes apontamentos, possibilidades de melhoria ou experimentos adicionais:

O processo de *Labelling*, ou rotulagem das *boundingboxes* toma tempo e exige cuidado e precisão na identificação dos alvos. Uma vez realizado, a imagem pode passar por processo de redimensionamento desde que as proporções sejam mantidas. Esse processo é válido quando se deseja baixar o tempo de processamento do treinamento da rede neural.

O *YOLO V4* mostrou uma adequada capacidade de detecção das faixas de pedestre incorrendo em erros de alguns *pixels* no desenho das caixas (*bounding box*). Foi constatada a capacidade de se identificar as faixas de pedestres em imagens oblíquas e de sensores orbitais com alta e baixa resolução. Outra constatação interessante é que a rede consegue identificar até mesmo faixas de pedestres que foram pintadas sem rigor geométrico. Contudo, esses erros de pintura são facilmente refletidos no cálculo da altura. Isso

foi constatado nas imagens 2 e 3. Sugere-se para trabalhos futuros pesquisar uma maneira de se utilizar rotações na identificação das faixas de pedestres. A rede mostrou uma melhor acurácia no desenho das caixas quando as faixas eram apresentadas totalmente na horizontal ou totalmente na vertical. Essa acurácia pode ser verificada no Erro em Delta das imagens 4, 5 e 7. As incoerências apresentadas nas imagens 8, 9 e 10 podem ser atribuídas ao *dataset*. Havia um número reduzido de imagens naquela faixa de altura de voo, logo, a realização de um treinamento com maior número de imagens nessas altura pode vir a solucionar esse problema.

Finalmente, para o cálculo da altura de voo foi utilizada uma fórmula simplificada do problema com o intuito de verificar se tal método vale a pena ou não de ser perseguido e melhorado. Afirma-se que os resultados preliminares alcançados indicam que sim. Para trabalhos futuros, sugere-se que a câmera tenha seus parâmetros melhor definidos e calibrados, testar a possibilidade de utilizar imagens que passaram por processo de correção geométrica. Isso pode levar a uma contagem diferente do número de *pixels* em largura e altura e, conseqüentemente, um valor mais acurado para a determinação da altura e voo.

Portanto, os resultados encontrados até aqui atingem o objetivo do trabalho que é propor uma alternativa para mitigar a negação ou interferência nos sinais GNSS e permitir a navegação aérea autônoma. Além disso, essa proposta pode ser aplicada em diferentes ramos da navegação ou orientação autônoma terrestre ou, até mesmo, submarina. Trata-se apenas de encontrar a melhor adaptação para a aplicação na realidade desejada.

#### 6. REFERÊNCIAS

- [1] David Poole, Alan Mackworth, and Randy Goebel. Computational intelligence: A logical approach. ISBN 978-0-19-510270-3, 2020.
- [2] BRASIL. Ica 100-12 regras do ar. *Comando da Aeronáutica. Departamento de Controle do Espaço Aéreo. Serviços de Tráfego Aéreo*, 2013.
- [3] Carlos Norberto Stumpf Bento. Vulnerabilidades da navegação por satélites, may 2014.
- [4] Gabriel Alves. Detecção de objetos com yolo – uma abordagem moderna, october 2020.
- [5] Peiyuan Jiang, Daji Ergu, Fangyao Liu, Ying Cai, and Bo Ma. A review of yolo algorithm developments. *The 8th International Conference on Information Technology and Quantitative Management The 8th International Conference on Information Technology and Quantitative Management*, pages 1066–1073, 2022.
- [6] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *CoRR*, abs/2004.10934, 2020.
- [7] BRASIL. *Manual Brasileiro de Sinalização de Trânsito ; 4*. Conran, Brasília, primeira edition, 2007.
- [8] Junyan Lu1, Chi Ma, Li Li, Xiaoyan Xing, Yong Zhang, Zhigang Wang, and Jiuwei Xu. A vehicle detection method for aerial image based on yolo. *Journal of Computer and Communications*, pages 98–107, 2018.
- [9] dji. *Manual do Usuário Phantom 4 PRO/PRO+*, volume 1.2. dji, 2017.