

IDENTIFICAÇÃO DE ARMAS DE FOGO EM TEMPO REAL UTILIZANDO UM DRONE AUTÔNOMO

Abstract— Violence is a global problem that affects millions of people. Technology can be a tool to increase the efficacy of security. The present research proposes the development of an autonomous Drone, that identifies, in real time, armed people, and bounding boxes the location of the weapon. For the gun recognition with moving aerial images, is used the de Kanade-Lucas-Tomasi method, linked to one pre-trained network of Matlab® for identifying people, reducing the frame of the analyzed video and, consequently, the processing time. At least, for the weapon identification, were used a dataset with 8.300 gun images and 12.617 without guns. Were compared 3 different training methods, to identify which is the best method. The methods use Convolutional Neural Network and they are AlexNet, VGG-16 (both with transfer learning) and classification using virtual words. The best result was obtained using AlexNet, with 87% of the filmed videos by the built drone.

Keywords— Violence, Drone, Weapon, Convolutional Neural Network, AlexNet, VGG-16, Kanade-Lucas-Tomasi.

Resumo— A violência é um problema mundial que afeta milhares de cidadãos. A tecnologia pode vir como uma forma de aumentar o nível de segurança da população. O presente artigo propõe o desenvolvimento de um drone autônomo que identifique pessoas armadas, em tempo real, realçando, por meio de caixas de seleção (retângulos gráficos), o local em que a arma está localizada. Para identificação de armas com imagens aéreas em movimento, foi usado o método de Kanade-Lucas-Tomasi. Este algoritmo foi atrelado à utilização de uma rede pré-treinada do Matlab® para identificação de pessoas, reduzindo o tamanho do *frame* do vídeo a ser analisado e, por consequência, do tempo de processamento. Por fim, para identificação da arma foi utilizado um *dataset* de 8.300 imagens com armas e 12.617 sem armas. Foram utilizados 3 diferentes métodos de treinamento para reconhecimento de objetos. Os métodos escolhidos utilizam do *Convolutional Neural Network* e são: AlexNet, VGG-16 (ambos com transferência de aprendizado) e classificação por palavras virtuais. O melhor resultado obtido foi o uso da rede AlexNet, com 87% de acerto nas filmagens feitas pelo drone construído.

Palavras-chave— Violência, Drone, Arma, *Convolutional Neural Network*, AlexNet, VGG-16, Kanade-Lucas-Tomasi.

1 Introdução

A violência é um problema mundial que afeta a vida de milhares de pessoas. No Brasil, cerca de 61.619 mortes violentas foram registradas no ano de 2016, número que se assemelha ao das mortes provocadas em Nagasaki, atingida pela bomba nuclear no período da segunda guerra mundial (Acayaba, C., 2017).

Diversos especialistas em segurança debatem a melhor forma de reduzir os casos de violência no mundo. Em uma entrevista do G1 com o diretor do Fórum Brasileiro de Segurança Pública, Renato Sérgio descreve que o problema da segurança não deve ser enfrentado apenas com o aumento do número de policiais, mas sim com inteligência. Ele aborda que é necessário investir em tecnologias que englobem a segurança nas ruas para permitir que o Estado possua mecanismos inteligentes de combate ao crime (Lima, 2017).

Visto os casos apresentados e como forma de investimento para medidas de segurança urbana, esse artigo propõe o reconhecimento de armas de fogo nas ruas por um quadricóptero autônomo. O equipamento verifica a presença de armamento numa determinada região e registra a localização da arma em tempo real. Além disso, o drone também é capaz de registrar a sua localização cartográfica no momento da identificação.

O presente trabalho comparou 3 formas de treinamento para reconhecimento de imagens, usando a metodologia conhecida como *Deep Learning* (Murphy, 2012). Esses métodos possuem em comum

o uso da *Convolutional Neural Network* (CNN). Os diferentes algoritmos de identificação de objetos são: Classificação por palavras virtuais e redes pré-treinadas, aplicando-se o *Transfer Learning* para AlexNet e VGG-16 (Liu, Wang, & Qiao, 2017). Além disso, utilizou-se do algoritmo de Kanade-Lucas-Tomasi (KLT) para reconhecimento do plano de fundo e por fim, a identificação da posição da arma utilizando-se segmentação de imagem para criar caixas de seleção (retângulos gráficos) nos *frames*.

2 Trabalhos relacionados

O uso das CNNs para identificação de cenas/objetos tem se tornado uma ferramenta muito usual nos dias atuais. Sua aplicação é muito vasta e já é usada tanto para aplicação de reconhecimento de rostos em redes sociais e aplicativos, como para identificação de anomalias em exames médicos ou reconhecimento de sinais de trânsito e pessoas para o controle de veículos autônomos.

Essas estratégias também estão sendo estudadas para a identificação de armas ou cenas de violência. Ruas (2017) realizou um projeto de identificação de armas com uso da Webcam do computador, usando o método *Tensorflow*. Nos trabalhos realizados por Demarty *et al.* (2012) e Lam *et al.* (2014), criou-se um programa para identificação de cenas de violência, baseada no áudio, imagens e movimentos de vídeos com o auxílio de *Support Vector Machine* (SVM) e o *Kernel model* para classificação. Bandyopadhyay *et al.* (2012) oferecem outra estratégia para identificação de armas, desenvolvendo um código que verifica as cores nos quadros e aplica

a correlação com uma imagem IR, utilizando *Discrete Wavelet Transform* (DWT).

Como forma de identificação do local da arma em um *frame*, Tiwari e Verma (2015) utilizaram uma associação da segmentação da imagem com o método *K-means*. Esse procedimento elimina pontos não nítidos dos vídeos recolhidos de câmeras de segurança. Por fim, para identificação de armas, utilizou-se o *Harris interest point detector* e *Fast Retina Keypoint* (FREAK). Grega *et al.* (2016) fizeram um projeto parecido, também utilizando câmeras de segurança, entretanto eles identificavam o plano de fundo da filmagem com o algoritmo de *Canny Edge*, criando diversas bordas nos objetos em movimento. A identificação da categoria nesse projeto foi feita com MPEG-7.

Trabalhos recentes de Olmosa *et al.* (2018), Lai e Maples (2017) utilizaram o *Region-based Convolutional Neural Networks Classifier* (R-CNN) para identificar armas de fogo em cenas de filme, também criando caixas de seleção para reconhecer a posição da arma.

O presente artigo mostra um diferencial em relação aos apresentados acima por buscar a identificação de armas de fogo por imagens aéreas. Além disso, o trabalho sugere a automação da rota do drone e as coordenadas geográficas no momento da identificação da imagem.

3 Inteligência Artificial & Deep Learning

A Inteligência Artificial (IA) está presente em diversas aplicações do mundo atual. Seu significado é amplo e sua atuação muito vasta. Alguns autores de importantes livros, relacionados ao tema, definem a IA como uma maneira nova, a ser explorada, para fazer computadores pensar (Haugeland, 1985). Já Bellman (1978), descreveu IA como a automação de atividades que são associadas com o pensamento humano, tais como tomada de decisão, solução de problemas e aprendizado (apud Russell e Norvig, 1995).

Independente da plenitude de significados que a IA possa ter, hoje ela é amplamente utilizada para resolução e otimização de problemas, planejamento, análise de dados, movimento e manipulação robótica, reconhecimento de imagens, entre outros.

Deep Learning é uma área da IA que utiliza técnicas para extração de informações diretamente dos dados, como imagens, textos ou som. O termo “*Deep*” está relacionado ao vasto número de camadas ocultas que a rede pode ter, enquanto que em outros tipos de redes neurais, utilizam-se usualmente entre 1 a 3 camadas (What Is Deep Learning?, 2017).

A seguir será feita uma breve explicação dos algoritmos e alguns conceitos importantes que foram utilizados no presente artigo.

3.1 Convolutional Neural Network

O tipo de rede mais utilizado no *Deep Learning* é o CNN, também conhecido como ConvNet. Esse método extrai as informações dos dados de entrada, como palavras virtuais, e usa camadas convolucionais 2D. Em termos gerais, quanto mais camadas ocultas, mais informações das imagens podem ser identificadas para o treinamento do sistema. A rede busca garantir o grau de mudança, escala e variância local das imagens, utilizando-se de campos, pesos compartilhados e subamostragem espacial ou temporal nas camadas ocultas (LeCun, Bégio e Haffner, 1988).

3.2 Classificação por palavras virtuais

Para esse método de treinamento extrai-se informações de um *dataset* de imagens. Os dados extraídos em formas de matrizes numéricas possuem informações que classificam a figura, como formas, traços ou conjunto de cores.

Cada objeto ou cena possui um aspecto, que é interpretado pelo computador como matrizes. Essas representações são classificadas como palavras virtuais. Objetos semelhantes apresentam palavras semelhantes, ou seja, elementos numéricos da matriz que se assemelham e são diferente de outros objetos. A partir disso é utilizada a regressão logística e classificação de conjuntos matriciais, para encontrar uma curva (função) que melhor classifica diferentes objetos.

3.3 AlexNet

É um modelo desenvolvido em uma competição de 2012 conhecida como *ImageNet Large-Scale Visual Recognition Challenge* (ILSVRC, 2018). Utiliza de uma rede de CNN em séries, com uma nova arquitetura, capaz de classificar 1000 diferentes categorias (The 9 Deep Learning Papers You Need To Know About, 2016). No treinamento feito com essa rede, foram utilizadas 25 camadas, sendo uma de entrada de dados, outra de saída e o restante representando o treinamento convolucional de alguns pixels extraídos nas imagens de entrada.

3.4 VGG-16Net

É outra arquitetura de CNN, assim como o AlexNet. Esse modelo possui 16 camadas ocultas e utiliza filtros convolucionais bem pequenos de 3 x 3 pixels (o menor tamanho, conforme o artigo de Simonyan e Zisserman (2015), para ter noção de esquerda/direita, cima, baixo e centro).

Esse método mostrou uma melhora significativa na identificação de imagens. Sua aplicação também foi designada para identificação de 1000 diferentes categorias, entretanto seu treinamento é bem mais lento do que o AlexNet (Simonyan e Zisserman, 2015).

3.5 Transfer Learning

É um método comumente utilizado em aplicações de *Deep Learning*. Ele consiste em utilizar uma rede já treinada - capaz de classificar um vasto número de diferentes objetos - para identificação de um novo conjunto de categorias. Isso permite que seu resultado seja obtido com um menor conjunto de imagens de treinamento e de forma mais rápida (Yosinski *et al.*, 2014).

3.5 Algoritmo de Kanade-Lucas-Tomasi (KLT)

Esse algoritmo foi apresentado em 1981 na 7th *International Joint Conference on Artificial Intelligence* e representa uma técnica de rastreamento de imagens. Esse método realiza, com o auxílio do algoritmo de Newton-Raphson, uma interação entre imagens, usando um gradiente de intensidade espacial. Esta técnica se mostra rápida por analisar um menor número de pontos das imagens. Sua desvantagem é que ao analisar menos pontos, o algoritmo torna a imagem menos nítida. Isso pode impactar em casos que se deseja rastrear objetos mais detalhados ou menores, que podem se tornar imperceptíveis (Lucas e Kanade, 1981).

4 Metodologia e Resultados

No presente trabalho, o drone construído (Figura 1) foi adaptado do projeto de Brokking (2018), com uso do microprocessador STM32F103C8T6 e adicionando o MS5611 (um barômetro para controle da altura). Além disso, colocou-se um Arduino com o GPS NEO-06 e, para outra opção de comunicação o NRF24L01 (um módulo de rádio frequência), como alternativa do controle remoto. A Figura 2 mostra o esquemático de blocos do projeto.

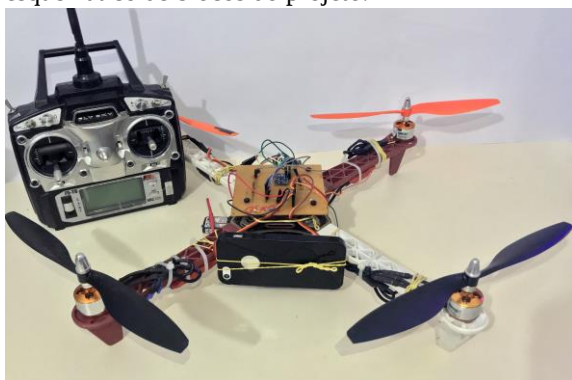


Figura 1 - Drone desenvolvido com todos os sensores, microcontrolador e NRF2401.

A ideia de construir o drone desde o início, ao invés de usar um comercial, foi para criar um equipamento de custo inferior e boa eficiência. Além disso foi um projeto didático que visava pôr em prática os conhecimentos de controle adquiridos na universidade.

No projeto o drone realiza um rota pré-determinada, especificada pelo operador através de

coordenadas de GPS. Para inserir esse percurso, são adicionados os pontos geográficos no Matlab®, que envia comandos para o receptor do Arduino fixado no drone via sinais de *Pulse Width Modulation* (PWM), transmitidos com o NRF2401.

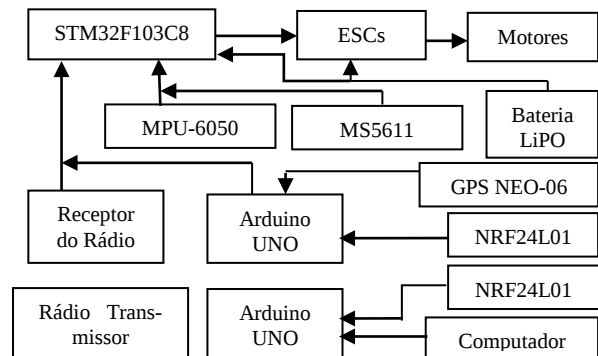


Figura 2. Diagrama em blocos do drone.

Para transmissão de imagens foi colocado um iPhone 5S, com câmera de 8 Mp e resolução de 3.264 x 2.448 pixels, na parte frontal do drone, que envia, em tempo real, a reprodução em vídeo da câmera para o computador, via IP, utilizando o aplicativo IpWebcam®. Em trabalhos futuros será necessário realizar um estudo para melhor captação de imagens com um custo menor. O iPhone foi escolhido por ser um equipamento disponível, de fácil transmissão de imagens em tempo real para o computador.

Os dados foram processados no Matlab® 2017a utilizando-se um computador com placa de vídeo GeForce GTX1060, processador AMD FX 8.350, memória 8 GB 1.600 MHz e sistema operacional Windows 10.

O procedimento para identificação de objeto consiste em analisar cada *frame* retirado da transmissão do drone. Caso seja reconhecida uma arma será identificada a sua posição na tela, com uso de retângulos, além da coordenada que foi realizada o reconhecimento.

Um dos desafios encontrados nesse projeto foi reduzir o tamanho da amostra da filmagem a ser analisada. A quantidade de informação contida na imagem está diretamente relacionada com o tempo de processamento da identificação de armas, por isso buscou-se um método para reduzir o tamanho das amostras que vão ser analisadas no programa.

A fim de resolver esse problema, antes de executar o algoritmo de identificação, extraiu-se o plano de fundo das filmagens, usando o método da diferença de *frames* (Singla, 2014). Entretanto, como o drone está em constante deslocamento, os movimentos do plano de fundo e objetos se misturavam, dificultando a aplicação desse artifício. Além disso, na filmagem a variação de altitude, perspectiva ou as próprias vibrações causadas pelo motor, aumentavam a dificuldade para extração do plano de fundo.

Para resolver esse problema utilizou-se um processamento de 30 *frames* por segundo, e a cada 5 segundos um novo plano de fundo é estimado, usando a mesma estratégia.

A partir disso, foi realizada uma análise da diferença entre os pixels dos *frames* (usando os comandos `detectSURFFeatures` e `extractFeatures`). Posteriormente induziu-se uma mudança geométrica entre as imagens (rotação/escala), e então uma compensação entre os *frames* (obtida com o `vision.PointTracker` que utiliza o algoritmo de KLT para correção entre as imagens, que retorna o erro entre os dois *frames*).

Conforme pode ser visto na Figura 3, a diferença entre 2 *frames* tem uma variação da posição entre os pixels. Conhecendo o deslocamento desses pontos, é possível ajustar os *frames*, aplicando-se a função *tfrom*. Então é determinada a diferença entre os quadros para identificação do plano de fundo. Os pixels que possuem um maior deslocamento são identificados como movimentos de objetos.

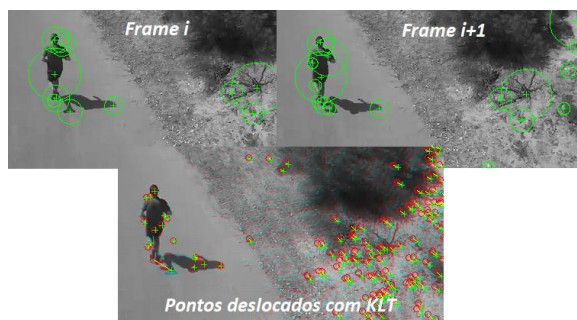


Figura 3. Aplicação do KLT para correção do deslocamento.

Para redução do tempo de análise e melhora dos resultados, utilizou-se uma rede pré-treinada da MathWorks® chamada *detectPeople*. Essa função retorna um retângulo marcando a posição das pessoas que estão nos *frames*. Esse retângulo foi ajustado para ter um tamanho definido, conforme altura da filmagem, possibilitando o reconhecimento de uma área maior ou menor. A ideia de identificar pessoas antes das armas é para reduzir o número de falsas alertas encontradas devido sombras de árvores ou outros objetos que possam se assemelhar com uma arma.

A partir do reconhecimento de pessoas, é aplicada a classificação na imagem para armas com 3 diferentes métodos: Classificação por palavras virtuais, AlexNet com *Transfer Network* e VGG-16 também com *Transfer Network*. Um fluxograma que mostra o procedimento realizado neste trabalho é disposto na Figura 4.

Para o treinamento dessas redes foi utilizado um *dataset* de 1.480 imagens de armas de fogo, retiradas do *site* de banco de dados de filmes (Guns, 2017). Essas figuras foram rotacionadas em 10° e 30°, nos sentidos horário e anti-horário. Também foram selecionadas 900 cenas de filmes, totalizando em 8.300 imagens recortadas. Adicionalmente, foram

utilizadas 12.617 imagens retiradas de Mazars (2017), representando outros itens como ambientes, pessoas praticando esportes ou segurando objetos.

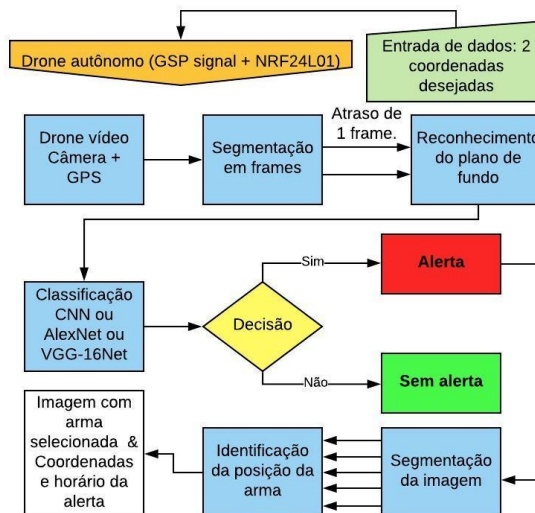


Figura 4. Fluxograma do algoritmo para identificação de armas de fogo.

Foi realizado o treinamento de cada uma das redes relatadas. Os resultados obtidos foram analisados em alguns vídeos de teste, extraídos do YouTube, de diversas cenas com e sem arma de fogo.

Para análise dos resultados separaram-se as classificações por *frames* em Verdadeiro Positivo (VP), que representa a identificação de arma em um quadro com arma; Falso Positivo (FP), simboliza a identificação de armas em um frame sem arma; Verdadeiro Negativo (VN), quando não é identificada uma arma em um frame sem armas; Falso Negativo (FN), quando não se identifica armas em frames com armas; Acerto Total (AT) e o tempo total de treinamento.

Usando o algoritmo de classificação por palavras virtuais extraíram-se 2.373.068 estruturas características diferentes a partir dos pixels das imagens que representam uma arma. Desse número, utilizando o agrupamento de *K-means*, disponível no Matlab®, formou-se 1000 conjuntos de palavras que definem uma arma. Essas palavras foram adotadas para o treinamento.

Para encontrar um padrão que melhor definisse a classificação do objeto foram utilizados diferentes funções de ativação com o *Support Vector Machine*. O melhor resultado foi obtido com o *Medium Gaussian*, com uma acurácia de 94,7 % das imagens de teste (30% do banco de dados total). Esse resultado foi obtido em 68,7633 minutos de treinamento.

Com o uso da *Transfer Network* usando AlexNet, modificaram-se as camadas 23 e 25, que representam a conexão dos 1000 diferentes objetos e a saída de informações destes, para as duas categorias que se deseja classificar (com arma e sem arma).

No treinamento da rede usou-se do *Stochastic Gradient Descent with Momentum* (SGDM), que

aplica uma redução dos passos em direção ao menor erro (Bishop, 2006). Esse método também evita a oscilação do mínimo global, adicionando um termo correspondente ao momento do vetor em direção ao valor mais otimizado (Murphy, 2012).

Foi definida uma taxa de treinamento inicial de 0,001, utilizado para atualizar os parâmetros da rede neural (pesos e *bias*). Nesse método é calculado o erro total e a variação do gradiente após a análise de cada interação. Esta é denominada como “era de treinamento” e corresponde a uma parcela do *dataset*.

O número de eras limite do treinamento foi definido como 20 entretanto, dentro de cada era, define-se um número de amostras aleatórias a serem treinadas. Esta amostragem foram de 100 imagens, com resolução 227x227 pixels (RGB). Posteriormente, foi realizado um treinamento que durou 25,08 minutos, com 5000 interações, e um acerto de 99% entre os testes das amostras.

Com uso do VGG-16, também foi adotado o SGDM com no máximo 20 eras com 100 amostras por era, com 0,001 de taxa de treinamento inicial. O processamento foi realizado com as imagens na resolução 32x32 RGB. Dessa vez também foi adicionado um decaimento para que a taxa de treinamento fosse multiplicada por um fator de 0,1 a cada 8 eras. Isso resultou em 4000 iterações, em 8,61 minutos, com 98% de acerto.

Os números de eras, amostras e a taxa de treinamento foram definidos como padrão para melhor comparação entre os métodos. Em pesquisas futuras será interessante a variação destes parâmetros para melhor obtenção de resultados.

A partir de todos os treinamentos, foram realizadas análises dos resultados para verificação da melhor resposta. Os resultados obtidos na Tabela 1 foram gerados a partir das imagens de teste, que correspondem a 30% de imagens aleatórias do banco de dados.

Tabela 1. Relação de acertos do banco de dados na identificação de Armas.

	VP	VN	FP	FN	AT	Tempo
(1)	46%	47%	3%	4%	93%	414,6s
(2)	39%	60%	0,7%	0,3%	99%	1504,9s
(3)	23%	66%	6%	5%	89%	516,8s

(1) - Classificação por palavras virtuais - Função (SVM Cúbica); (2) - AlexNet (Transfer Network); (3) - VGG-16 (Transfer Network)

A partir dos resultados da tabela verificou-se que a melhor função para classificação de armas foi a AlexNet, possuindo um acerto de cerca de 99% do banco de dados.

Após a identificação da existência de arma de fogo no *frame*, buscou-se encontrar a possível localização da arma no quadro. Para isso, usou-se da função do Matlab® *trainRCNNObjectDetector*, que se apropria do *deep learning* para detectar regiões com objetos.

No seu funcionamento são selecionadas as posições das armas, criando-se Regiões Retangulares de Interesse (ROIs) em todas as imagens do banco de dados (Figura 5), utilizando da ferramenta *Image Labeler*. Neste método, todas as imagens que estão dentro da caixa de seleção são tratadas como armas e todo o restante, sem arma. Como parâmetros da rede, utilizou-se o SGDM com 128 amostras de treinamento, 0,001 como taxa de treinamento inicial, que é multiplicada por 0,1 a cada 10 eras, sendo o número máximo de eras igual a 100.



Figura 5. Separação manual das informações para identificação de armas de fogo.

Esse método de classificação identifica áreas como positivas conforme a Equação 1, para $C \geq 0,5$ (valor definido nas opções de treinamento).

$$C = \frac{\text{Área}(A \cap B)}{\text{Área}(A \cup B)} \quad (1)$$

Nessa equação A e B são regiões selecionadas no treinamento e nas imagens de teste, respectivamente.

Esse método resultou em 99% de acerto das imagens de teste, com 110,92 minutos de treinamento.

Para verificar os resultados de uma maneira ampla, utilizaram-se 5 vídeos disponíveis no YouTube, com imagens aéreas captadas com drone. Isso resultou em 288 segundos de vídeo e 6.433 frames. Realizando a análise e contagem manual dos frames, verificou-se que cerca de 14% dos frames analisados tinham ao menos uma arma. Os vídeos escolhidos foram gravados com diferentes drones, em alturas variadas e com qualidades distintas de filmagem.

Os resultados obtidos são mostrados na Tabela 2. Para a melhora dos resultados considerou-se como frames VP aqueles que possuíam precisão fosse superior a 85% na classificação. Já para a identificação de ROIs, foi considerado VP o frame com arma de fogo quando a acurácia fosse superior a

85% e, ao menos, uma caixa de seleção estivesse correta.

Tabela 2. Relação de acertos na identificação de Armas com vídeos aéreos do YouTube.

	VP	VN	FP	FN	AT
(1)	3%	80%	6%	11%	83%
(2)	6%	82%	4%	8%	88%
(3)	2%	81%	5%	12%	83%
(4)	1%	84%	2%	13%	85%
(5)	5%	83%	3%	9%	88%
(6)	1%	80%	6%	13%	81%

(1) - Classificação por palavras virtuais - Função (SVM Cúbica); (2) - AlexNet (Transfer Network); (3) - VGG-16 (Transfer Network); (4) - Classificação por palavras virtuais com Caixas de Seleção (CS); (5) - AlexNet com CS; (6) - VGG-16 com CS

A partir dos resultados obtidos, pode-se ver que o maior número de FP relatados foram regiões de sombras, ou com presença de árvores, conforme mostrado na Figura 6, que representa uma identificação correta da presença de uma arma, e uma sombra como uma arma. A Figura 7 mostra o sucesso no reconhecimento de uma arma com uso do drone.



Figura 6. Identificação falsa e verdadeira de arma de fogo



Figura 7. Identificação de arma de fogo com imagem aérea.

A função que apresentou a melhor resposta foi novamente a de AlexNet, com 88% de acerto. Além disso observou-se que, praticamente em todos os *frames* que se identificou uma arma com o AlexNet também foi identificada o ROI.

Após a análise dos resultados com os vídeos do YouTube foram realizados testes com o uso do drone construído e descrito na Seção 4. O experimento foi feito para realizar uma simples rota, identificando as pessoas presentes e se há ou não a existência de uma arma de fogo no local, identificando as coordenadas destes pontos.

A Figura 8 expõe a rota do drone e os pontos que foram identificadas armas em um dos voos. É importante ressaltar que, nesses registros de coordenadas, existe uma imprecisão, dada pelo fabricante, de 2 até 6m pelo GPS.

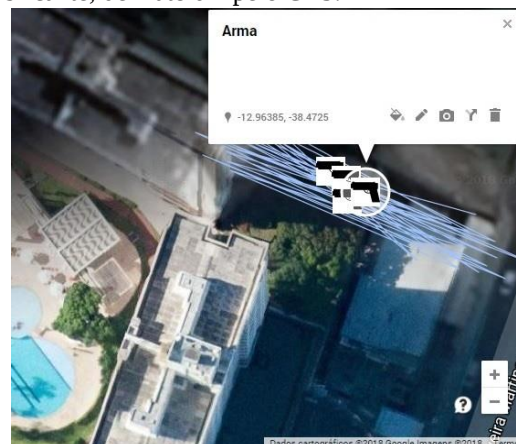


Figura 8. Coordenadas registradas pelo drone.

Com a realização dos testes, de acordo com a altura das filmagens, foi feita uma escolha dos parâmetros para identificação de pessoas – o tamanho da área que seleciona a pessoa, como pode ser visto na Figura 9. A partir disso, foi analisada qual a rede treinada que apresentou um melhor resultado (Tabela 3). Foram registrados 4674 *frames* sendo que cerca de 16% desses quadros possuíam uma arma. Novamente, só foram considerados VP quando a identificação apresentava uma precisão superior a 85%.

Tabela 3. Relação de acertos na identificação de Armas com o drone.

	VP	VN	FP	FN	AT
(1)	1%	80%	4%	15%	81%
(2)	4%	82%	2%	12%	86%
(3)	3%	81%	3%	13%	84%
(4)	3%	78%	6%	13%	81%
(5)	4%	83%	1%	12%	87%
(6)	2%	79%	5%	14%	81%

(1) - Classificação por palavras virtuais - Função (SVM Cúbica); (2) - AlexNet (Transfer Network); (3) - VGG-16 (Transfer Network); (4) - Classificação por palavras virtuais com Caixas de Seleção (CS); (5) - AlexNet com CS; (6) - VGG-16 com CS

A partir dos resultados obtidos na Tabela 3, observou-se que a melhor rede foi a AlexNet, obtendo-se um percentual de acerto de 87%. Na maioria das detecções onde foi reconhecida uma arma no *frame*, também foi marcada uma caixa de seleção mostrando o local em que foi identificado o objeto. Novamente, exibindo-se todos os retângulos detectados, foram observados FP que associavam as caixas de seleção a sombras ou outros pontos mais escuros, que se assemelhavam a uma arma.

A Figura 9 mostra o funcionamento do sistema de classificação. Na esquerda são rastreadas as pessoas que estão presentes no *frame* e, na direita, a identificação da arma. As Figuras 9, 10 e 11 mostram

regiões que foram encontradas armas corretamente. Na Figura 10 é identificada erroneamente uma sombra como arma e na Figura 12, o cabelo de uma menina, como uma arma.



Figura 9. Identificação das pessoas e da arma no vídeo feito pelo drone.

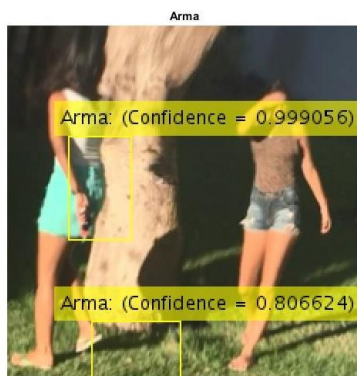


Figura 10. Identificação correta e incorreta de uma arma de fogo no vídeo feito pelo drone.



Figura 11. Identificação correta de uma arma de fogo no vídeo feito pelo drone.



Figura 12. Identificação incorreta de uma arma de fogo no vídeo feito pelo drone.

4 Conclusão

A tecnologia pode ser utilizada para reduzir os casos de violência espalhados pelo mundo e apoiar medidas de segurança, como o controle de armamento. O presente artigo mostrou e comparou diferentes métodos existentes, na identificação de armas de fogo e associou a aquisição de imagens em tempo real por um drone construído.

O uso do algoritmo *Kanade-Lucas-Tomasi* possibilitou a extração de imagens em movimento entre os *frames*. Isso reduziu o número de pixels a serem analisados e por consequência, aumentou a velocidade de processamento do programa. Foi observado que atrelar o uso desse algoritmo com a identificação de pessoas em um ambiente, reduziu o número de verdadeiro negativo que ocorre muito em regiões escuras ou com sombras.

Houve uma atenção especial na redução de falsos positivos, para reduzir o número de alertas incorretas ao operador do sistema proposto. Observa-se que quanto maior o número de falsos alertas, menor o interesse do operador em ficar monitorando o sistema.

Seguindo esse princípio, obteve-se como melhor resposta das redes treinadas, tanto dos testes com vídeos do YouTube, quanto da filmagem com o drone montado, a rede AlexNet. Com esta rede, obteve-se um percentual de acertos de cerca de 88% das imagens analisadas no Youtube, com o menor número de falsos alertas e maior número de alertas verdadeiros. Quando se utilizou, em série, a rede VGG-16 para identificação de caixas de seleção, a eficiência da rede permaneceu a mesma, entretanto, apesar do número de alertas verdadeiros ter reduzido 1%, o número de alertas falsos também reduziu em 1%. Para os vídeos em tempo real com o quadricóptero o acerto total dessa rede foi de 87%.

Comparando a acurácia obtida com o treinamento, usando a rede AlexNet, na classificação de armas, e outros trabalhos que também identificavam armas com uso da CNN, observou-se um acerto total parecido. No trabalho de Ruas (2017), obteve-se um acerto de 90%, Lai e Maples (2017) 89%, Tiwari e Verma (2015) com 84,26%, e o melhor resultado dos artigos analisados foi de Grega, *et al* (2016) com 94,93% de acerto. Levando em conta que o presente artigo, por se tratar de imagens aéreas com muitas vibrações, constante movimento da filmagem, qualidade inferior das imagens coletadas, quando comparado com os outros trabalhos, concluiu-se que um acerto de 87 % dos vídeos coletados foram satisfatórios.

A rede AlexNet se mostrou como a melhor solução para identificação de armas de fogo. Apesar dos bons resultados, significativas melhoras devem ser feitas para ser utilizada com sucesso em aplicações de segurança pública ou privada.

Para trabalhos futuros pretende-se investir em um bom drone, com maior estabilidade e qualidade

da filmagem a fim de dedicar maior tempo ao reconhecimento de imagens aéreas. Além disso, para a melhora do banco de dados, seria interessante realizar filmagens aéreas com pessoas armadas, circulando em ambientes abertos. Em suma, ainda é interessante procurar outros métodos de treinamento para reconhecimento de objetos e modificar os números de eras, amostras e a taxa de treinamento, a fim de verificar se há ou não uma melhora nos resultados.

Referências Bibliográficas

- Acayaba, C. (30 de Outubro de 2017). Brasil registra o maior número de assassinatos da história em 2016; 7 pessoas foram mortas por hora no país. Acesso em 27 de Dezembro de 2017, disponível em G1.Globo: <https://g1.globo.com/sao-paulo/noticia/brasil-registra-o-maior-numero-de-homicidios-da-historia-em-2016-7-pessoas-foram-assassinadas-por-hora-no-pais.ghtml>
- Bandyopadhyay, S., Datta, B., & Roy, S. (2012). Identifications of concealed weapon in a Human Body. *International Journal of Scientific & Engineering Research*, III(11).
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Nova York: Springer.
- Brokking, J. (2018). Project YMFC-32 - The STM32 quadcopter. Fonte: Brokking.net: http://www.brokking.net/ymfc-32_main.html
- Demarty, C.-H., Penet, C., & Gra, G. (27 de Novembro de 2012). The MediaEval 2012 Affect Task: Violent Scenes. HAL archives-ouvertes.fr.
- Grega, M., Matiolański, A., Guzik, P., & Leszczuk, M. (2016). Automated Detection of Firearms and Knives in a CCTV Image. *Sensors*, pp. 47-63.
- Guns. (2017). Fonte: IMFDB: <http://www.imfdb.org/>
- Haugeland, J. (1985). *Artificial Intelligence: The Very Idea*. Londres: The MIT Press.
- ILSVRC. (2 de Julho de 2018). Challenges. Fonte: ImageNet Large Scale Visual Recognition Challenge (ILSVRC): <http://www.image-net.org/challenges/LSVRC/>
- Lai, J., & Maples, S. (2017). Developing a Real-Time Gun Detection Classifier.
- Lam, V., Le, D.-D., Phan, S., Satoh, S., & Duong, D. A. (2014). NII-UIT at Violent Scenes Detection Affect Task. *MediaEval*.
- LeCun, Y., Bengio, Y., & Haffner, P. (1988). Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*.
- Lima, R. S. (Outubro de 2017). Segurança pública. (G1.Globo, Entrevistador)
- Liu, J., Wang, Y., & Qiao, Y. (2017). Sparse Deep Transfer Learning for Convolutional Neural Network. *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*, pp. 2245-2251.
- Lucas, B. D., & Kanade, T. (1981). An Iterative Image Registration Technique with an Application to Stereo Vision. *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, 674-679.
- Mazars, G. (20 de Novembro de 2017). CV Datasets on the web. Fonte: CV Papers: <http://www.cvpapers.com/datasets.html>
- Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. Cambridge, Massachusetts: The MIT Press.
- Olmosa, R., Tabik, S., & Herrera, F. (2018). Automatic Handgun Detection Alarm in Videos Using Deep Learning. *Neurocomputing*, 66-72.
- Ruas, M. S. (2017). Detecção de objetos com ameaça à integridade física em tempo real: um desenvolvimento com CNN. Salvador, Bahia, Brasil.
- Russell, S. J., & Norvig, P. (1995). *Artificial Intelligence. A Modern Approach*. New Jersey: Prentice-Hall, Inc.
- Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *ICLR 2015*.
- Singla, N. (15 de Novembro de 2014). Motion Detection Based on Frame Difference Method. *International Journal of Information & Computation Technology*, IV(15), 1559-1565.
- The 9 Deep Learning Papers You Need To Know About. (24 de Agosto de 2016). Acesso em 27 de Dezembro de 2017, disponível em <https://adeshpande3.github.io/adeshpande3.github.io/The-9-Deep-Learning-Papers-You-Need-To-Know-About.html>
- Tiwari, R. K., & Verma, G. K. (2015). A Computer Vision based Framework for Visual Gun Detection using Harris Interest Point Detector. *Elsevier*, 703-712.
- Yosinski, J., Clune, J., Bengio, Y., & Lip, H. (2014). How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems* 27.