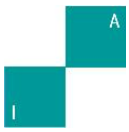
Inteligencia Artificial, 24(67) (2021), 40-50
doi: 10.4114/intartif.vol24iss67pp40-50



INTELIGENCIA ARTIFICIAL

http://journal.iberamia.org/

A Temporal Fusion Approach for Video Classification with Convolutional and LSTM Neural Networks Applied to Violence Detection

Jean Phelipe de Oliveira Lima[1] and Carlos Maurício Seródio Figueiredo[2]

[1]Escola Superior de Tecnologia, Universidade do Estado do Amazonas, Manaus-AM, Brazil
jpdol.eng16@uea.edu.br

[2]LSI - Laboratório de Sistemas Inteligentes, Universidade do Estado do Amazonas, Manaus-AM, Brazil
cfigueiredo@uea.edu.br

Abstract In modern smart cities, there is a quest for the highest level of integration and automation service. In the surveillance sector, one of the main challenges is to automate the analysis of videos in real-time to identify critical situations. This paper presents intelligent models based on Convolutional Neural Networks (in which the MobileNet, InceptionV3 and VGG16 networks had used), LSTM networks and feedforward networks for the task of classifying videos under the classes "Violence" and "Non-Violence", using for this the RLVS database. Different data representations held used according to the Temporal Fusion techniques. The best outcome achieved was 0.91 and 0.90 of Accuracy and F1-Score, respectively, a higher result compared to those found in similar researches for works conducted on the same database.

Keywords: Applications of AI, Deep Learning, Intelligent Video Processing, Violence Detection.

1 Introdução

A modernização da sociedade e das tecnologias urbanas tendem a resultar nas chamadas cidades inteligentes, que são caracterizadas pela capacidade de implementação de tecnologias de comunicação [1], com o objetivo de proporcionar o maior nível de integração e automatização de serviços. Para isso, são desenvolvidas inúmeras soluções para monitoramento de ambientes, que envolvem uso de sensores, câmeras, entre outros, a fim de se obter relatórios, apoio à tomada de decisão, tomada de ação, etc. Esse alto nível de monitoramento gera um grande acúmulo de dados, fenômeno conhecido como *Big Data* [2], que abre espaço para a denominada Ciência de Dados buscar gerar valor a partir de análises inteligentes sobre dados [3].

Na busca pela maior automatização de tarefas possível, análises inteligentes sobre dados são realizadas fazendo-se uso de técnicas de Aprendizado de Máquina [4], utilizando a grande quantidade de dados para identificação de padrões, previsão de fenômenos, etc. No setor da segurança, especificamente, o monitoramento é realizado via câmeras de vigilância, o que gera enormes quantidades de imagens por segundo. Normalmente, a supervisão desses dados é realizada por humanos, o que é uma tarefa extremamente difícil, visto o grande volume de dados gerados em tempo real por vídeos de segurança. Uma solução para isso é a utilização de técnicas de *Deep Learning* [5] para o processamento automático de vídeos, como acontece em [6] e [7], para identificação, por exemplo, de situações de risco e/ou incidentes, tais como: acidentes de trânsito; incêndio; assalto; violência; atentado ao pudor; etc.

ISSN: 1137-3601 (print), 1988-3064 (on-line)

Percebe-se a necessidade de uma forma de análise mais eficiente no sistema de monitoramento de câmeras de vigilância. A análise humana pode ser imprecisa e não rápida o suficiente para que gere uma ação imediata. Propõe-se, então, um sistema que analise automaticamente um vídeo e seja capaz de identificar uma situação de violência e, de maneira muito mais ágil, apoiar tomada de decisão e de ação. Aplicação esta, muito útil em ambientes que demandam um forte monitoramento, como vias públicas, shoppings, estações de metrô e penitenciárias, por exemplo, onde, uma vez identificada uma cena de violência, entidades responsáveis podem ser brevemente acionadas, evitando incidentes ainda maiores.

Em [8], é apresentada uma solução baseada em redes neurais recorrentes e convolucionais para o problema de detecção de violência em vídeos com resultados superiores aos apresentados em [9] e [10], que utilizam apenas redes convolucionais, indicando que a melhor abordagem para a tarefa em questão está na utilização conjunta de redes recorrentes e convolucionais. Este trabalho apresenta como contribuição a proposta de aplicação de ConvLSTM (*Convolutional* LSTM), isto é, um modelo baseado em CNNs (Redes Neurais Convolucionais) e LSTMs (*Long Short-Term Memory*), além do emprego e avaliação de técnicas de representação de vídeo baseada em *Temporal Fusion* (Fusão Temporal), o que proporcionou melhor desempenho em relação aos trabalhos da literatura. Na Seção 2 serão apresentados trabalhos relacionados ao contexto deste, em seguida, na Seção 3, serão informados os materiais e métodos utilizados, seguido da apresentação e análise dos resultados obtidos, na Seção 4. Por fim, na Seção 5, serão expostas as considerações finais e perspectivas de trabalhos futuros.

2 Trabalhos Relacionados

Técnicas de *Deep Learning*, sobretudo Redes Neurais Convolucionais, têm obtido êxito em pesquisas relacionadas a visão computacional, como mostra [11]. Isso acontece pela definição de CNNs, que são baseadas no sistema de visão humano, sendo assim bastante aplicada à problemas de classificação de imagens e identificação de conteúdo nas mesmas, como apresentam [7] e [12], por exemplo.

Trabalhos utilizando CNNs para a tarefa de detecção de violência em vídeos têm sido frequentes. Em [13], é apresentada a construção de redes convolucionais para a classificação binária de vídeos violentos ou não violentos, enquanto, em [10], os autores propõem soluções baseadas em *Transfer Learning* a partir de modelos, também de CNNs, treinados com a base de dados ImageNet [14]. Outros trabalhos, apresentam ainda a combinação de outros modelos com redes neurais convolucionais, o que é o caso de [9], que apresenta uma solução baseada em *Hough Forest* e CNNs.

Por outro lado, redes neurais recorrentes, possuem boa performance ao tratar de séries temporais [15]. Essa capacidade pode ser explorada na tentativa de aumentar o desempenho de modelos, já que vídeos possuem características temporais entre seus *frames*. Esse processo ocorre em [8], que utiliza redes convolucionais e LSTM para classificação de vídeos de violência.

Portanto, propõe-se um modelo de *deep learning* baseado em redes neurais convolucionais e redes neurais recorrentes do tipo LSTM, resultando em uma rede ConvLSTM, para exploração tanto de características espaciais como temporais em vídeos, de modo a contribuir na tarefa proposta em classificação automática de vídeos para detecção de cenas de violência. Este trabalho se diferencia de trabalhos relacionados por utilizar as técnicas mencionadas e ainda fazer uso de *Transfer Learning* para a composição dos modelos propostos, além de aplicar técnicas de representação de vídeos, baseada em *Temporal Fusion*, não encontrada em trabalhos que abordam a mesma tarefa proposta por este.

3 Materiais e Métodos

Esta seção descreve os materiais e métodos utilizados para realização dos experimentos conduzidos para a tarefa de detecção de violência em vídeos. As subseções são divididas em: Dados Experimentais, que apresenta e avalia a base de dados utilizada; Tarefa de Aprendizado de Máquina, que informa como o Aprendizado de Máquina será empregado em face à base de dados apresentada; *Temporal Fusion*, que aborda as técnicas de representação dos dados de entrada; Proposição de Modelos, que apresenta os modelos de Aprendizado de Máquina propostos para realização da tarefa em questão; e Avaliação de Desempenho, que apresenta as métricas utilizadas para avaliação dos modelos propostos.

3.1 Dados Experimentais

Foi utilizada a base de dados *Real Life Violence Situations Dataset* (RLVS-2019) [16] que consiste na coleção de 2.000 exemplos de vídeos curtos, com tempo médio de duração de $5,4s$ e amostragem média de $29,5 fps$, separados em duas classes: "Violência"e "Não Violência". Os exemplos são divididos igualmente entre as classes, o que implica em uma base balanceada, reduzindo assim as chances de *overfitting*. Os exemplos da classe "Violência" são amostras de vídeos capturados por câmeras de segurança, trechos de filmes e vídeos reais encontrados no *Youtube*. A classe "Não Violência"é composta também por vídeos reais e por trechos de filmes que apresentam situações cotidianas normais, como prática de esportes, conversas, abraços, caminhada, etc. A Figura 1 apresenta, por meio de sequências de *frames*, um exemplo contido em cada uma das classes da base de dados utilizada.

Figura 1: Amostras de vídeos contidos na base de dados RLVS-2019.

Analisando os *frames* individualmente, há uma certa dificuldade de entendimento do contexto da cena em questão, o que dificultaria a tarefa de detecção de violência a partir de imagens estáticas. O terceiro *frame* do exemplo da classe "Violência", por exemplo, se analisado individualmente, não há como afirmar se representa ou não uma cena violenta, entretanto, quando analisados os *frames* em sequência, ficam mais evidentes as características que diferenciam cada uma das classes. Dessa forma, para a tarefa de classificação de vídeos quanto às duas classes mencionadas, a utilização de modelos de *Deep Learning* para a análise de *frames* temporalmente dependentes é potencialmente uma solução para o problema.

3.2 Tarefa de Aprendizado de Máquina

A base de dados apresentada será utilizada para realização de um aprendizado de máquina supervisionado com a tarefa de classificação binária dentre as classes "Violência"e "Não Violência". Tal tarefa será executada em duas etapas: fase de treino e fase de validação. Para isso, será utilizada a técnica de Validação Cruzada do tipo *K-Fold*, onde $K = 10$. Isto significa que a base de dados será subdividida em 10 partes iguais, também chamadas de *splits*, em que, a cada iteração, uma *split* é utilizada como conjunto de validação e as demais são usadas como conjunto de treino. Isso se repete 10 vezes, isto é, até que todas as *splits* tenham sido utilizadas como conjunto de validação.

Durante a fase de treino de cada iteração, os exemplos do conjunto de treino (resultado da união de todos os subconjuntos exceto o conjunto de validação) serão apresentados às redes neurais, cujas arquiteturas serão apresentadas na subseção 3.4, em um total de 50 épocas, a fim de que haja os ajustes de seus parâmetros treináveis, gerando, assim, modelos inteligentes. Para garantir o poder de generalização dos modelos, na fase de validação de cada iteração, exemplos do conjunto de validação, não utilizados na fase de treino, serão apresentados aos modelos e suas saídas obtidas serão comparadas às saídas desejadas para cada exemplo, para assim avaliar seu desempenho, de acordo com métricas apresentadas na subseção

3.5. Após todas as rodadas de treino e validação terem sido executadas, as métricas de desempenho serão consideradas a partir da média aritmética e desvio padrão do desempenho obtido na validação cruzada.

3.3 *Temporal Fusion*

Um dos principais desafios da utilização de vídeos como atributos preditores de modelos de aprendizado de máquina é a forma com que esses dados serão representados. A preocupação principal é de manter as características temporais nesses dados sem que a quantidade de *frames* utilizados inviabilize o treino desses modelos.

Uma abordagem para a representação de vídeos é a *Temporal Fusion* que, de acordo com [17], apresenta várias formas de associação entre os *frames* dos vídeos. Essas diferentes formas de representação dos dados são enumeradas a seguir e ilustradas pela Figura 2.

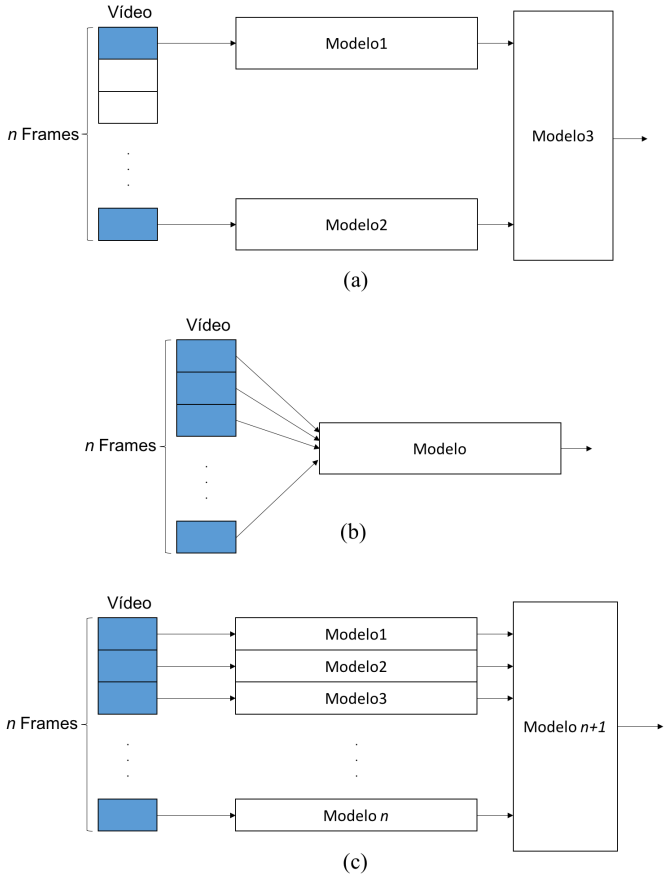


Figura 2: Arquitetura das técnicas de *Temporal Fusion* para representação de um vídeo por meio associação de *frames*. (a) *Late Fusion*. (b) *Early Fusion*. (c) *Slow Fusion*.

1. ***Late Fusion*:** Consiste na utilização do primeiro e do último *frame* de cada exemplo, a fim de analisar o início e o fim da ação representada em cada vídeo. Cada um dos *frames* utilizados por essa abordagem são primeiramente processados individualmente pelos modelos, cujas saídas são, em seguida, processadas de maneira relacionada, como mostra a Figura 2.a. A técnica possui este nome, *Late Fusion* (Fusão Tardia), por tratar da fusão, após o processamento, das características dos *frames* com maior distância temporal entre si, isto é, que delimitam o início e fim de uma cena.

2. ***Early Fusion*:** Consiste no processamento, por parte de um modelo, de *frames* concatenados de cada vídeo. A Figura 2.b ilustra esta técnica de representação dos dados de entrada. O nome *Early*

Fusion (Fusão Precoce), é dado a esta técnica pelo fato de que a fusão dos dados acontece antes mesmo de qualquer *frame* ser processado.

3. ***Slow Fusion*:** Consiste na técnica que processa os *frames* do vídeo separadamente por modelos, cujas saídas são, em seguida, processadas de maneira relacionada, como ilustra a Figura 2.c. Dá-se o nome de *Slow Fusion* (Fusão Lenta) a esta técnica, por fazer a associação temporal de diversos *frames* após passarem por um determinado processamento.

Em [17], os autores apresentam, ainda, a técnica denominada *Single Frame*, que seleciona apenas um *frame* de cada vídeo como seu representante. Essa técnica não foi abordada por este trabalho, pois trata o problema como uma tarefa de classificação de imagens. Apenas técnicas para representação de vídeos foram utilizadas.

Para as abordagens *Slow Fusion* e *Early Fusion* é necessário que sejam selecionados *frames* do vídeo para representá-lo. É fundamental que a quantidade de *frames* escolhidos para essas abordagens não seja tão grande, a ponto de inviabilizar o treino, tampouco tão pequeno, a ponto de que haja perdas significativas de características temporais do vídeo. Como os vídeos da base de dados possuem, em média, cerca de $5s$ de duração, decidiu-se utilizar 2 *frames* por segundo, conforme sugerido em [17] após testes empíricos, para os vídeos que possuem duração média, ou seja, 10 *frames* por vídeo. Esse número de *frames* foi aplicado a todos os vídeos da base, dessa forma há uma variação de *frames* por segundo amostrados, porém mantendo um número fixo de *frames* por vídeo. Nos vídeos que possuem tempos de duração diferentes da média, os *frames* foram extraídos igualmente espaçados em relação ao tempo.

3.4 Proposição de Modelos

A subseção 3.3 demonstrou como os vídeos serão apresentados aos modelos de aprendizado de máquina que, como será visto a seguir, são baseados em técnicas de *Deep Learning*. Em [17], os autores, além de proporem as técnicas de Temporal Fusion para apoio a representação dos dados em tarefas de classificação de vídeos, sugerem que os modelos utilizados sejam baseados em Redes Neurais Convolucionais, como forma de extração de características das situações encontradas nos vídeos, seguido de camadas *feedfoward* de neurônios perceptron, que realizam o reconhecimento de padrão a partir das características extraídas nas camadas anteriores. Essa abordagem é muito utilizada por várias aplicações de *Deep Learning*, sobretudo à tarefas de visão computacional. Entretanto, vídeos possuem uma característica não encontrada em imagens estáticas: a dependência temporal. Ou seja, a sequência temporal entre *frames* pode ser melhor explorada por uma técnica apropriada para extração desse tipo de característica. Portanto, camadas LSTM serão utilizadas com este propósito, por representarem o estado da arte em extração de características temporais em sinais.

A estratégia, então, baseia-se em ConvLSTMs, ou seja, utilizar as camadas LSTM para analisar as características temporais contidas em suas entradas, que por sua vez representam características espaciais extraídas pelas camadas convolucionais. Assim, a estratégia é: apresentar os dados de entrada às camadas convolucionais, para que ocorram as extrações de características espaciais dos *frames*; seguido de uma camada LSTM que extrai características temporais a partir dos dados resultantes das camadas convolucionais; e finalmente realizar uma classificação dos padrões por meio da utilização de camadas *feedfoward* de neurônios perceptron. A Figura 3 ilustra essa visão geral, em blocos, da arquitetura ConvLSTM dos modelos que serão construídos, divididos entre as técnicas de *Temporal Fusion* apresentadas.

Pela Figura 3 é possível verificar tanto o fluxo dos diferentes tipos de redes neurais, respeitando a sequência CNN – LSTM – *Feedfoward*, e também a forma com que os *frames* dos vídeos serão submetidos aos modelos, de acordo com as técnicas de *Temporal Fusion*. Para cada uma das arquiteturas apresentadas nessa figura, fez-se diversos testes empíricos para composição dos parâmetros e hiperparâmetros das redes neurais. Para os blocos de redes neurais convolucionais, serão utilizadas as camadas convolucionais das redes canônicas: MobileNet; InceptionV3; e VGG16 por meio da técnica de *Transfer Learning*, em que serão utilizados pesos resultantes do treino com a base de dados ImageNet, ou seja, a técnica de *Fine-Tuning* não será aplicada e apenas as camadas finais, adicionadas aos modelos, terão seus pesos ajustados. Além disso foi construída uma rede neural simples com a parâmetros e hiperparâmetros apresentados na Figura 4 e treinada a partir da base de dados RLVS-2019.

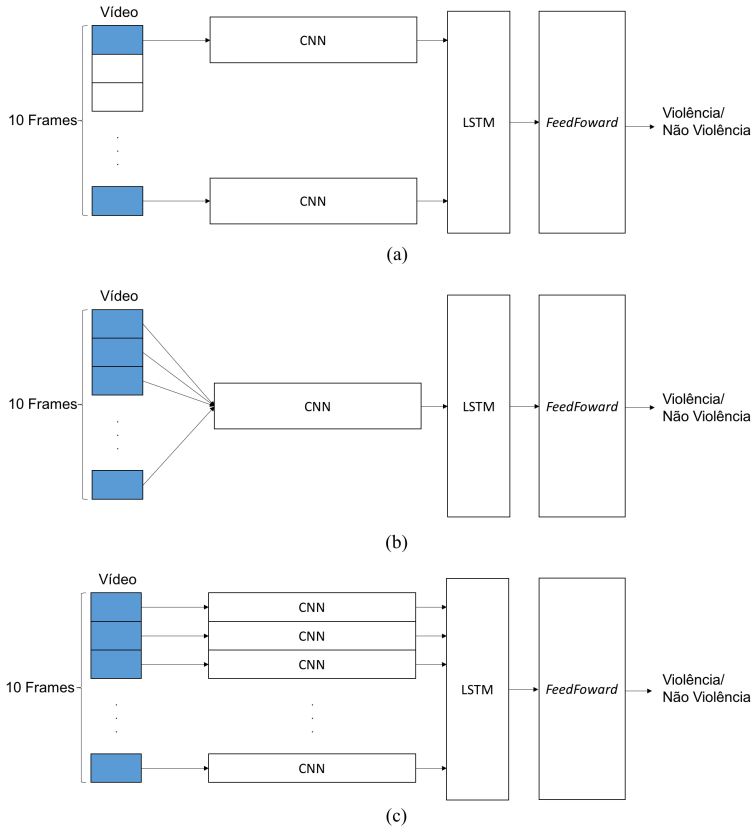


Figura 3: Apresentação das propostas de arquiteturas: (a) baseada em *Late Fusion*; (b) baseada em *Early Fusion*; e (c) baseada em *Slow Fusion*.

Na Figura 4 os parâmetros para as camadas convolucionais são, respectivamente, número de núcleos de convolução, tamanho de *kernel* e função de ativação. Para as camadas de *pooling* o parâmetro informado representa a dimensão da matriz de *pooling* e o parâmetro informado para as camadas de *batch normalization* representa o *momentum*. Foi utilizado, para todos os modelos o otimizador "Adam".

As redes mencionadas serão testadas como blocos convolucionais para cada uma das arquiteturas apresentadas na Figura 3 juntamente com o bloco de saída, composto pelas camadas finais dos tipos LSTM e *feedfoward*. Essas camadas finais foram empiricamente definidas conforme ilustrado na Figura 5, em que os parâmetros definidos para as camadas *feedfoward* são, respectivamente, número de neurônios perceptron e função de ativação, para as camadas LSTM o parâmetro informado representa a quantidade de neurônios LSTM, enquanto para as camadas *dropout* representa a taxa de normalização.

Portanto, para cada uma das técnicas de *Temporal Fusion* utilizadas serão testados 4 modelos, baseados nas junções dos 4 blocos convolucionais citados com o bloco de saída cujas camadas são apresentadas na Figura 5.

3.5 Avaliação de Desempenho

Os resultados obtidos serão avaliados de acordo com as métricas de desempenho Acurácia e F1-Score. A acurácia é dada por: $Acurácia = \frac{TP+TN}{TP+FP+TN+FN}$, em que TP representa o número de verdadeiros positivos; TN o número de verdadeiros negativos; FP o número de falsos positivos; e FN o número de falsos negativos. Essa métrica representa a proporção de classificações corretas dentre o total de classificações realizadas, trazendo uma noção intuitiva do desempenho do modelo e, neste caso, confiável, por tratar de um modelo treinado a partir de uma base de dados balanceada. Todavia, será utilizada também a métrica F1-Score, denotada por: $F1\text{–}Score = \frac{2 \cdot Precisão \cdot Revocação}{Precisão + Revocação}$ que consiste em uma métrica

mais robusta que a acurácia por representar a média harmônica entre a precisão e a revocação, sendo estas denotadas por: $Precisão = \frac{TP}{TP+FP}$ e $Revocação = \frac{TP}{TP+FN}$.

Como a validação do modelo utiliza a técnica *K-Fold*, os resultados apresentados serão a Acurácia e F1-Score médio e seus respectivos desvios padrões após terem sido executadas todas as iterações da Validação Cruzada. Além disso, serão apresentadas as métricas temporais: Tempo de Treino (em segundos); e Tempo Médio de Execução (em milissegundos), para que os modelos sejam avaliados quanto a performance computacional.

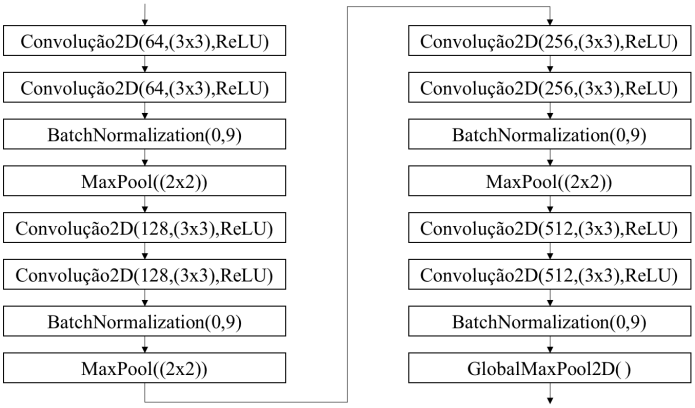


Figura 4: Rede neural de arquitetura simples construída.

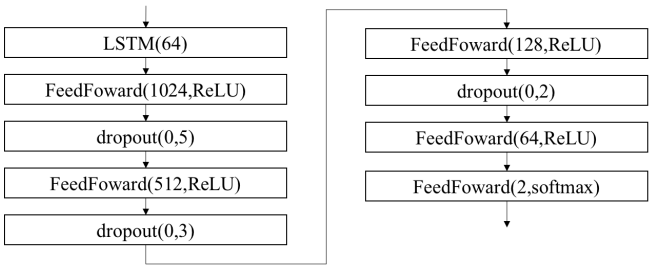


Figura 5: Camadas finais dos modelos propostos que serão utilizadas após as camadas convolucionais.

4 Resultados e Discussões

Esta seção apresenta os resultados obtidos a partir dos experimentos realizados de acordo com a metodologia proposta para a classificação de vídeos para detecção de situações de violência. O treino e validação de todos os modelos foram baseados em Validação Cruzada do tipo *K-Fold*, que dividiu a base em 10 *splits* de teste. Para os modelos que utilizaram *Transfer Learning* apenas o bloco de saída teve seus pesos neurais ajustados, enquanto os modelos que usaram a rede simples tiveram todas as suas camadas treinadas. As subseções apresentam os resultados obtidos da fase de validação e estão divididas em: *Late Fusion*; *Early Fusion*; *Slow Fusion*, que apresentam e avaliam o desempenho obtido pelos modelos propostos para cada uma dessas técnicas de *Temporal Fusion*; e, por fim, Comparação com Trabalhos Relacionados, que lista o desempenho alcançado pelos modelos propostos em face aos do estado-da-arte.

4.1 *Late Fusion*

A Tabela 1 apresenta os resultados obtidos pelos modelos baseados na técnica *Late Fusion*, em que o primeiro e o último *frame* de cada vídeo da base são apresentados, individualmente, para os blocos convolucionais relacionados na tabela, seguido do bloco de saída que contém camadas LSTM e *feedfoward*.

Os resultados demonstraram-se confiáveis com o fato de a acurácia e F1-Score apresentarem valores semelhantes, indicando que o treino dos modelos foi conduzido adequadamente.

Tabela 1: Resultados obtidos pelos modelos baseados em *Late Fusion*.

Bloco convolucional	Acurácia Média	Desvio Padrão de Acurácia	F1-Score Médio	Desvio Padrão de F1-Score	Tempo de Treino (s)	Tempo Médio de Execução (ms)
Simples	74,01%	1,01	73,40%	0,98	550,00	54,18
MobileNet	80,23%	0,24	80,32%	0,21	250,00	25,02
InceptionV3	76,40%	0,67	76,17%	0,72	450,00	39,68
VGG16	88,79%	0,18	88,50%	0,20	555,00	52,24

Para essa abordagem, o modelo com bloco convolucional VGG16 demonstrou melhores resultados quanto a Acurácia e F1-Score, superando em cerca de 12% o modelo com bloco convolucional InceptionV3 e 14% o modelo com bloco convolucional simples. O resultado superior a 88% de acurácia e F1-Score, apesar de competitivos a modelos do estado da arte, ainda sugerem possibilidades de melhoria de resultado pelas outras técnicas de *Temporal Fusion*, visto que *Late Fusion* é a técnica, dentre as abordadas, que menos contribui para a extração de características temporais, uma vez que são utilizados apenas o frame inicial e o final de cada um dos exemplos.

Quanto ao tempo de treino e execução dos modelos, o modelo com bloco convolucional MobileNet se mostrou mais eficiente que todos os demais, o que era esperado justamente pela arquitetura em questão ser mais simplificada que as demais. Entretanto as métricas de desempenho do modelo baseado em MobileNet foram inferiores a do modelo baseado em VGG16.

4.2 *Early Fusion*

A Tabela 2 apresenta o desempenho dos modelos com os blocos convolucionais relacionados na tabela e com *Temporal Fusion* do tipo *Early Fusion*, em que os 10 *frames* que representam cada vídeo são concatenadas e apresentados ao modelo. Percebe-se que, com exceção do modelo com bloco convolucional MobileNet, todos os outros modelos sofreram uma certa perda de desempenho em relação aos modelos de mesmo bloco convolucional referentes à técnica *Late Fusion*. Isso indica, em termos gerais, que pelo fato de a concatenação dos dados ser realizada antes da extração de características espaciais, o próprio bloco convolucional pode perder algumas características temporais dificultando a tarefa da camada LSTM e, por consequência reduzindo o desempenho dos modelos.

Tabela 2: Resultados obtidos pelos modelos baseados em *Early Fusion*.

Bloco convolucional	Acurácia Média	Desvio Padrão de Acurácia	F1-Score Médio	Desvio Padrão de F1-Score	Tempo de Treino (s)	Tempo Médio de Execução (ms)
Simples	71,71%	1,23	70,90%	1,10	525,00	48,12
MobileNet	85,30%	0,90	84,92%	0,89	230,00	28,11
InceptionV3	72,24%	0,14	71,40%	0,20	479,00	45,59
VGG16	84,12%	0,31	83,21%	0,41	520,00	50,80

Assim como na abordagem de *Late Fusion*, o modelo baseado em MobileNet se apresentou mais eficiente na avaliação temporal, sendo assim o modelo com melhor desempenho para ambas as avaliações: de desempenho (avaliado pela Acurácia e F1-Score) e de performance computacional (avaliada pelos tempos de treino e de execução dos modelos).

4.3 *Slow Fusion*

Os últimos modelos testados foram baseados na técnica de *Slow Fusion*, que foi realizada utilizando 10 *frames* de cada vídeo como entrada de 10 blocos de convolução idênticos. Ao todo foram testadas 4

diferentes composições para os blocos de convolução, como apresenta a Tabela 3 com seus respectivos resultados quanto as métricas utilizadas.

Tabela 3: Resultados obtidos pelos modelos baseados em *Slow Fusion*.

Bloco convolucional	Acurácia Média	Desvio Padrão de Acurácia	F1-Score Médio	Desvio Padrão de F1-Score	Tempo de Treino (s)	Tempo Médio de Execução (ms)
Simples	72,33%	0,25	70,14%	0,33	650,00	61,32
MobileNet	91,02%	0,67	90,89%	0,61	290,00	29,92
InceptionV3	76,66%	0,21	76,00%	0,21	500,00	49,84
VGG16	89,08%	1,20	88,71%	0,99	630,00	56,47

Observa-se que todos os modelos com blocos convolucionais em que foi utilizado *Transfer Learning* obtiveram os melhores desempenho para esta abordagem, o que era esperado, pois *Slow Fusion* apresenta a maior quantidade de extração de características espaciais individuais para os *frames* analisados, o que contribui para o melhor desempenho da camada LSTM na tarefa de extração de características temporais nos seus dados de entrada advindos dos blocos convolucionais. Essa característica justifica o uso da abordagem *Slow Fusion* pelo modelo com melhor resultado geral do trabalho, sendo este o modelo com bloco convolucional MobileNet que obteve 91,02% de acurácia e 90,89% de F1-Score. Além disso, o modelo baseado em MobileNet se manteve como o modelo com menor tempo de treino e execução, por tratar-se de um modelo mais leve que os demais.

4.4 Comparação com Trabalhos Relacionados

Em [16], os autores, além de organizarem a base de dados RLVS, propõem diversos modelos para a tarefa de classificação de vídeos para detecção de cenas de violência. A comparação mais adequada para o trabalho apresentado deve ser de um trabalho relacionado que apresente uma solução para a mesma tarefa e, preferencialmente, utilize a mesma base de dados. Neste caso, a utilização da mesma base de dados para a comparação de resultados é ainda mais importante pelo fato de que as principais bases de dados utilizados em outros artigos, inclusive referências citadas na Seção 2, trazem estudos referentes à base de dados consideravelmente menores, como é o caso da *The Hockey Dataset* [18] com 1000 imagens; *Movie Dataset* [18] com 200 imagens; e *Violent-Flow Dataset* [19] com 246 imagens. A utilização dessas bases de dados pequenas, ou com imagens relativamente parecidas (como é o caso da *The Hockey Dataset*), pode implicar em resultados tendenciosos. Por outro lado, a base RLVS, proposta por [16], contém as imagens dessas bases mencionadas e ainda inclui exemplos novos. Dessa forma, considera-se a base RLVS mais apropriada como *benchmark* para a tarefa em questão.

Portanto, a Tabela 4 apresenta os melhores resultados de [16] e deste trabalho, que propõem modelos baseados em *Deep Learning* para a tarefa em questão. O modelo que representa este trabalho é o baseado na abordagem *Slow Fusion* com bloco convolucional MobileNt, por ter apresentado os melhores resultados quanto às métricas analisadas. O modelo com melhores métricas apresentado em [16], por sua vez, para a base de dados RLVS foi baseado na arquitetura da VGG16 utilizando técnicas de *fine-tuning* seguido de camadas LSTM.

Tabela 4: Comparação dos resultados obtidos com os encontrados em [16].

Modelo	Acurácia
[16]	88,20%
Proposto	**91,02%**

A Tabela 4 evidencia o aumento em desempenho, medido pela métrica acurácia, entre o trabalho que é estado da arte para a tarefa em questão e o modelo apresentado neste trabalho. O modelo proposto apresentou 3% a mais de acurácia que o trabalho relacionado, o que indica que este representa o estado da arte para a tarefa, validando assim as abordagens de *Temporal Fusion*, em especial *Slow Fusion*, como

técnicas de representação de dados para análise de vídeos por contribuir para a competitividade dos resultados dos modelos apresentados para a tarefa de detecção de violência em vídeos curtos.

Muitos trabalhos relacionados, como é o caso de [10]; [9]; [8]; e [13], apresentam soluções para a mesma tarefa que este trabalho com desempenho próximo a 100%, entretanto utilizando a base de dados *Movie Dataset* [20], que apresenta cenas de filmes, também divididas em duas classes ("Violência"e "Não Violência"), no entanto com apenas 100 exemplos para cada classe. Essa falta de dados gera um dificuldade de generalização dos modelos, como prova [16] ao treinar e validar um modelo a partir da base *Movie Dataset* e, em seguida, validar com a base RLVS. A acurácia obtida para a validação com o *Movie Dataset* foi de aproximadamente 0,99 enquanto para a base RLVS foi de 0,75. Portanto, a base de dados RLVS se faz mais adequada para realização da tarefa, inclusive por conter os exemplos do *Movie Dataset*, e por isso escolhida para condução dos experimentos apresentados.

5 Considerações Finais

O trabalho apresentou o teste e validação de diversos modelos de *Deep Learning*, baseados em ConvLSTM, para a tarefa de detecção de violência em vídeos. Os modelos foram divididos por abordagens de *Temporal Fusion*, em que a técnica de *Slow Fusion* apresentou o melhor resultado com 0,91 de acurácia e F1-Score quando utilizado o bloco convolucional MobileNet. Resultado este superior a modelos do estado da arte que utilizaram para treino e validação a base de dados RLVS, também utilizada por este trabalho. Além de um melhor desempenho na classificação, o resultado de usar o modelo MobileNet com *Slow Fusion* torna a solução mais viável para plataformas de menor poder computacional, por tratar-se de um modelo leve, o que é uma vantagem para aplicações de vigilância. Essa viabilidade pode ser evidenciada pelo baixo tempo de treino e execução dos modelos com bloco convolucional MobileNet em comparação com os demais modelos em cada um dos testes das diferentes abordagens de *Temporal Fusion*.

Como propostas de trabalhos futuros, sugere-se utilizar outras técnicas de representação de vídeos, inclusive aumentando ou diminuindo o número de *frames* utilizados neste trabalho e também testes com entrada de imagens no domínio da frequência podem ser realizados (apenas o domínio do espaço foi abordado neste trabalho). Outra variável que pode ser explorada é o bloco convolucional, onde podem ser testadas inúmeras outras arquiteturas de CNNs. Outras técnicas de *Deep Learning* podem ser testadas em busca do aumento dos resultados alcançados, como a utilização de Redes Neurais Convolucionais 3D, que contemplam núcleos de convolução de 3 dimensões, que pode contribuir para extração de características, ao mesmo tempo, espaciais e temporais.

Referências

[1] M. C. Weiss, R. C. Bernardes, and F. L. Consoni, "Cidades inteligentes como nova prática para o gerenciamento dos serviços e infraestruturas urbanos: a experiência da cidade de Porto Alegre," *urbe. Revista Brasileira de Gest. Urbana*, vol. 7, pp. 310 – 324, 12 2015.

[2] I. A. T. Hashem, V. Chang, N. B. Anuar, K. Adewole, I. Yaqoob, A. Gani, E. Ahmed, and H. Chiroma, "The role of big data in smart city," *International Journal of Information Management*, vol. 36, no. 5, pp. 748 – 758, 2016.

[3] D. M. Blei and P. Smyth, "Science and data science," *Proceedings of the National Academy of Sciences*, vol. 114, no. 33, pp. 8689–8692, 2017.

[4] K. Faceli, A. Lorena, J. Gama, and A. Carvalho, *Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina*. LTC, 1st ed., 2011.

[5] Y. LeCun, Y. Bengio, and G. Hinton, *Deep Learning*. Springer, 2015.

[6] L. Calderoni, D. Maio, and S. Rovis, "Deploying a network of smart cameras for traffic monitoring on a "city kernel"," *Expert Systems with Applications*, vol. 41, no. 2, pp. 502 – 507, 2014.

[7] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. Baik, "Convolutional neural networks based fire detection in surveillance videos," *IEEE Access*, vol. 6, pp. 18174–18183, 2018.

[8] S. Sudhakaran and O. Lanz, "Learning to detect violent videos using convolutional long short-term memory," in *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–6, 2017.

[9] I. Serrano, O. Deniz, J. L. Espinosa-Aranda, and G. Bueno, "Fight recognition in video using hough forests and 2d convolutional neural network," *IEEE Transactions on Image Processing*, vol. 27, no. 10, pp. 4787–4797, 2018.

[10] A. S. Keçeli and A. Kaya, "Violent activity detection with transfer learning method," *Electronics Letters*, vol. 53, no. 15, pp. 1047–1048, 2017.

[11] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: A brief review," *Computational Intelligence and Neuroscience*, 2018.

[12] A. B. Sargano, X. Wang, P. Angelov, and Z. Habib, "Human action recognition using transfer learning with deep representations," in *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 463–469, 2017.

[13] P. Zhou, Q. Ding, H. Luo, and X. Hou, "Violent interaction detection in video based on deep learning," *Journal of Physics: Conference Series*, vol. 844, p. 012044, jun 2017.

[14] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[15] T. Guo, Z. Xu, X. Yao, H. Chen, K. Aberer, and K. Funaya, "Robust online time series prediction with recurrent neural networks," in *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 816–825, 2016.

[16] M. M. Soliman, M. H. Kamal, M. A. El-Massih Nashed, Y. M. Mostafa, B. S. Chawky, and D. Khattab, "Violence recognition from videos using deep learning techniques," in *2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)*, pp. 80–85, 2019.

[17] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[18] E. B. Nievas, O. D. Suarez, G. B. Garc, and R. Sukthankar, "Violence detection in video using computer vision techniques," *International conference on Computer analysis of images and patterns*, pp. 332–339, 2011.

[19] T. Hassner, Y. Itcher, and O. Kliper-Gross, "Violent flows: Realtime detection of violent crowd behavior," *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1–6, 2012.

[20] E. B. Nievas, O. D. Suarez, G. B. Garcia, and R. Sukthankar, "Movies fight detection dataset," in *Computer Analysis of Images and Patterns*, pp. 332–339, Springer, 2011.