Geração de Mapas de Dificuldade de Vídeos de Datasets

Generation of Difficulty Maps of Videos from Datasets

Katharina Akemi Ikeda Rosa *, Silvio Ricardo Rodrigues Sanches†

RESUMO

A avaliação do desempenho de um algoritmo de detecção de mudanças consiste basicamente em executá-lo para segmentar vídeos de um *dataset* e, em seguida, comparar os resultados com um *ground truth*. Essa estratégia, apesar de muito utilizada por pesquisadores da área, não considera algumas informações que podem ser úteis para melhorar o processo de avaliação. O nível de dificuldade para classificar pixels de um quadro de vídeo é uma dessas informações. Essa medida, que representa o esforço esperado para que um algoritmo classifique determinado pixel, é armazenada em uma estrutura chamada mapa de dificuldade. Neste trabalho, utiliza-se esses mapas como base para o desenvolvimento de métodos que são capazes de (i) estimar o nível de dificuldade de um vídeo e (ii) selecionar um subconjunto representativo de vídeos de um determinado *dataset*. Os resultados mostraram que o subconjunto selecionado por meio do método desenvolvido é representativo, pois apresentou nível de dificuldade similar ao do conjunto original.

Palavras-chave: detecção de mudanças. avaliação de algoritmos. *dataset*.

ABSTRACT

Evaluating a change detection algorithm performance basically comprises running it to segment the videos from a dataset and then comparing the results with a ground truth. This strategy, despite being used by researchers, ignores some information that can be useful to improve the evaluation process. The level of difficulty in classifying a video frame's pixels is one such piece of information. This measure represents the expected effort for an algorithm to classify a particular pixel. A structure called difficulty map store the level of difficulty of a frame. In this work, we used these maps as a basis for our methods that are used for (i) estimating the level of difficulty of a video and (ii) selecting a representative subset of videos from a dataset. The results showed the subset of videos selected through our method is representative, as it presented a level of difficulty similar to that of the original dataset.

Keywords: change detection. algorithm evaluation. dataset.

1 INTRODUÇÃO

A avaliação de desempenho de um algoritmo de detecção de mudanças é essencial para que se demonstre a superioridade de um novo algoritmo quando comparado aos algoritmos do estado-da-arte. Normalmente, a avaliação consiste em executar o algoritmo para segmentar vídeos de um *dataset* para que os resultados sejam comparados com um *ground truth*. *Ground truths* são conjuntos de quadros rotulados manualmente de forma que seja possível identificar o resultado ideal da segmentação. A partir desse resultado são calculadas métricas que representam o desempenho do algoritmo (GOYETTE et al., 2012).

* Departamento Acadêmico de Computação; katharina@alunos.utfpr.edu.br.
† Departamento Acadêmico de Computação; silviosanches@utfpr.edu.br.

Quando vários algoritmos utilizam o mesmo *dataset* e as mesmas métricas na etapa de avaliação é possível comparar seus desempenhos, pois os resultados são obtidos a partir das mesmas ferramentas e métodos. Essa forma de avaliar desempenho, ainda que bastante utilizada pelos pesquisadores da área, não considera algumas informações relevantes que podem ser úteis para comparar algoritmos. O nível de dificuldade para classificar determinado pixel, por exemplo, pode identificar as regiões de um quadro nas quais é difícil diferenciar o que é elemento de interesse (região que ocorre mudança) do que é plano de fundo (região estática). Essas regiões são pixels pertencentes aos vídeos dos *datasets* em que muitos algoritmos, mesmo os mais eficientes, falham ao classificar seus pixels (SANCHES; OLIVEIRA et al., 2019). Uma estrutura chamada mapa de dificuldade armazena os níveis de dificuldade de cada pixel de cada quadro de um vídeo.

Utilizando um mapa de dificuldade, apresentam-se neste trabalho abordagens para estimar o nível de dificuldade de um vídeo e para selecionar um subconjunto representativo de vídeos de um determinado *dataset*. Considera-se que um subconjunto representativo quando os vídeos que o compõe tem potencial de avaliação similar ao conjunto de vídeos completo do *dataset*. Em outras palavras, avalia-se vários algoritmos utilizando esses dois conjuntos (todos os vídeos do *dataset* e subconjunto representativo) e a ordem dos algoritmos é a mesma nas duas avaliações, quando esses algoritmos são ordenados considerando seus desempenhos. Os subconjuntos representativos podem ser utilizados para melhorar a eficiência de modelos de aprendizagem. Como não há redundância nos vídeos selecionados, a possibilidade de o modelo tomar decisões com base em ruídos é pequena. Além disso, o conjunto selecionado possui menos vídeos que o conjunto completo. Isso facilita a avaliação de versões preliminares de algoritmos, que podem ser avaliadas com maior rapidez.

2 MÉTODO

O primeiro passo para geração do método que seleciona vídeos representativos de um *dataset* consiste em criar uma estrutura chamada mapa de dificuldade, que é a informação base dos métodos aqui apresentados. Esse mapa deve ser gerado utilizando os resultados de diversos algoritmos, preferencialmente os que representam o estado-da-arte. Com esses resultados, é possível identificar o nível de dificuldade de um pixel contando quantos algoritmos classificaram incorretamente esse pixel. Dessa forma, para cada quadro, gera-se um mapa de dificuldade correspondente, que é responsável por armazenar esses valores. A primeira proposta para geração de mapas de dificuldade foi apresentada no trabalho de iniciação científica voluntária (PIVICT 2019/2020), desenvolvido pela mesma autora desta pesquisa. Neste trabalho, o processo foi aperfeiçoado e a estrutura foi utilizada para selecionar um subconjunto representativo de um *dataset*.

2.1 Geração do mapa de dificuldade

A geração dos mapas de dificuldade utiliza a seguinte abordagem. Um algoritmo de detecção de mudança gera uma máscara $S \in \{0,1\}^{l \times c}$ como resultado (valores normalizados), onde 1 é o rótulo dos pixels da região em que houve mudança, 0 é rótulo dos pixels do plano de fundo e $l \times c$ (linha $\times$ coluna) é a resolução do quadro do video. Os rótulos do *ground truth* $G \in [0,1]^{l \times c}$ são 0 (pixels do plano de fundo) e 1 (pixels dos elemento de interesse). A matriz $R \in \{0,1\}^{l \times c}$ indica os pixels que pertencem à região de interesse do *ground truth*, que são apenas os pixels rotulados como elemento de interesse ou plano de fundo. A matriz R é necessária porque um *ground truth* pode conter rótulos para identificar regiões como "sombras", por exemplo, que não são utilizadas pela abordagem na geração dos mapas.

Para gerar um mapa, executa-se vários algoritmos para segmentar vídeos de um *dataset* e, posteriormente, as máscaras S que contêm os resultados dos algoritmos são comparadas com o *ground truth* G para identificar os pixels classificados incorretamente. O nível de dificuldade de um pixel é dado pelo número de algoritmos que classificaram incorretamente o pixel. Para cada quadro de cada vídeo de um *dataset* é gerado um mapa de dificuldade, que é definido como $D \in [0,1]^{l \times c}$ e armazena o nível de dificuldade para classificar cada pixel.

2.2 Método para estimar o nível de dificuldade de um vídeo

O mapa de dificuldade gerado pode ser utilizado para estimar os níveis de dificuldade, denominado L, de um conjunto de vídeos. Esse nível de dificuldade está relacionado com o esforço necessário para classificar os pixels dos quadros desse vídeo. A versão inicial deste método foi desenvolvida no trabalho de iniciação científica voluntária (PIVICT 2019/2020), desenvolvido pela autora desta pesquisa. Os experimentos necessários para refinamento do método e para a otimização de parâmetros foram realizados neste trabalho.

Dado o número de pixels válidos Nvp para o j^{th} quadro de um *ground truth* G

$$Nvp_j = \sum_{i=1}^{l*c} p(i) \qquad p(i) = \begin{cases} 1, & \text{if } R_{(i)} == 1 \\ 0, & \text{if } R_{(i)} == 0 \end{cases} \tag{1}$$

o nível de dificuldade L de uma sequência de quadros k pode ser obtido de acordo com a equação

$$L(k) = \sum_{j=start}^{end} \sum_{i=1}^{N_{vp}} f \times \frac{d(i,j,D_k)}{N_{vp_j}} \tag{2}$$

onde $start$ é o quadro inicial, end é o quadro final, D_k é o mapa da sequência de quadros k, f é uma constante para reduzir o tamanho da escala dos valores que representam o nível de dificuldade (definido empiricamente como 0,1) e $d(i,j,D_k)$ é o nível de dificuldade armazenado do pixel i do quadro j do mapa de dificuldade D_k.

2.3 Métodos para selecionar vídeos representativos de um *dataset*

Uma vez estimados os níveis de dificuldade L de cada vídeo do *dataset* é possível utilizar essa medida para avaliar se os vídeos originais formam um conjunto equilibrado no que se refere aos seus valores de L. Vídeos com níveis similares de dificuldade podem produzir o mesmo valor de desempenho quando um mesmo algoritmo segmenta seus quadros, ao passo que vídeos com valores de L distintos podem produzir diferentes valores para o desempenho desse mesmo algoritmo. Um *dataset* em que os vídeos são selecionados de forma que seus valores de L representem uma escala de dificuldade pode tornar mais precisa a avaliação e a comparação de algoritmos com desempenhos similares.

2.3.1 Método baseado na média dos níveis de dificuldade

Para selecionar os vídeos representativos do *dataset*, a primeira abordagem adaptou um método desenvolvido em um trabalho de iniciação científica desenvolvido dentro do grupo de pesquisa em que a aluna está inserida. Nesse caso, o mapa de dificuldade gerado não é utilizado para como parte do método. Na estratégia, dado um *dataset* completo $D_{(v)}$, os v vídeos são divididos em g grupos. Para gerar o subconjunto $D_{(g-1,v)}$, utiliza-se clusterização considerando os níveis de dificuldade dos vídeos L, para distribuir os v vídeos em $g-1$ grupos. Para gerar o subconjunto $D_{(g-2,v)}$, o algoritmo considera os níveis L para distribuir os v vídeos em $g-2$ grupos

e, assim, sucessivamente até que o subconjunto $D_{(1,v)}$ contenha os v vídeos em um único grupo. Nesta etapa, os grupos dentro de um mesmo subconjunto contêm vídeos com L similares ou contêm um único vídeo.

O passo seguinte seleciona em cada grupo dentro de um mesmo subconjunto, o vídeo que possua o L mais próximo da média dos valores de L. Utilizando o *dataset* completo $D_{(v)}$, o método executa os n algoritmos, que são os mesmos utilizados para obter o nível de dificuldade L de cada vídeo, para segmentar os v vídeos e calcula os desempenhos ($F1$) de cada um dos algoritmos. Em seguida, calcula-se o $F1$ dos mesmos n algoritmos, segmentando apenas os vídeos do subconjunto $D_{(v-1)}$, $D_{(v-2)}$, $D_{(v-3)}$ e assim, sucessivamente. Esse processo gera a matriz de desempenhos $P \in [0,1]^{v \times n}$ que contêm os $F1$ referentes aos desempenhos de cada algoritmo calculados usando todos os vídeos do *dataset* ($D_{(v)}$) e calculados utilizando apenas os vídeos de cada subconjunto. Depois de ordenados os valores de $F1$ dentro de cada coluna da matriz P, o método obtêm a matriz de distâncias $M \in [0,1]^{(v-1)\times(n-1)}$, que contêm as distâncias entre os valores de $F1$ calculados utilizando um mesmo subconjunto de vídeos D. Cada coluna da matriz M armazena um conjunto de distâncias. Para cada um desses conjuntos, encontra-se o valor máximo armazenado em cada coluna de M para definir o vetor de tolerância T. Quando o valor de L de determinado vídeo for maior que o valor de tolerância, o vídeo é incluído no conjunto final.

2.3.2 Método baseado em mapa de dificuldade

O segundo método utilizado nesta pesquisa para selecionar vídeos representativos foi desenvolvido em conjunto com um trabalho de mestrado do mesmo grupo. Ao contrário do primeiro, o segundo método utiliza mapas de dificuldades para auxiliar a seleção dos vídeos. Inicialmente, os vídeos foram ordenados conforme seus valores de L, para gerar um vetor de distâncias. Remove-se, então, os *outliers* desse vetor utilizando o método interquartil de intervalo (RUSSEL; COHN, 2013), o que gera o vetor M'. Um threshold t que representa a mediana de M' dividido por 2 é calculado. Considerando o conjunto ordenado, um vídeo é incluído no subconjunto Rp se a diferença entre L do próprio vídeo e L do vídeo posterior é maior que t, conforme mostrado no Algoritmo 1.

Algorithm 1 Geração do subconjunto Rp

Entradas: V (número de vídeos do *dataset*), v (conjunto de vídeos do *dataset*), L (níveis de dificuldade)
Saída: Rp (subconjunto representativo dos vídeos do *dataset*)

1: $Rp = 0$
2: **for** i=2 **to** V **do**
3: $\quad M_{(i)} = abs(L_{(i)} - L_{(i-1)})$
4: $M' = M - outliers$
5: $t = (median(M'))/2$
6: **for** i=1 **to** $V - 1$ **do**
7: $\quad$ **if** $M_{(i)} > t$ **then**
8: $\quad\quad Rp = Rp + v(i)$

3 RESULTADOS E DISCUSSÕES

Ambos os métodos apresentados foram avaliados neste trabalho. O método 1, no entanto, mostrou resultados consideravelmente inferiores ao método 2. Por esse motivo, os resultados da sua avaliação foram omitidos neste relatório.

3.1 Geração dos mapas de dificuldade dos vídeos do CDNet 2014

Neste trabalho, para facilitar a avaliação do método, foram utilizadas as máscaras S disponíveis no *site* do *dataset* CDNet 2014 (UNIVERSITÉ DE SHERBROOKE, 2019). No site são disponibilizados resultados de algoritmos que obtiveram os melhores desempenhos entre os que utilizaram os vídeos do CDNet 2014 para avaliação.

As métricas taxa de falsos positivos PFR, taxa de falsos negativos FNR, precisão Pr, revocação Re, especificidade Sp, percentual de erros de classificação PWC e F-measure $F1$ representam os desempenhos dos algoritmos. A $F1$ tem sido a métrica mais utilizada pelos pesquisadores (SANCHES; SEMENTILLE et al., 2021). As máscaras S de 30 algoritmos foram utilizadas para gerar os mapas e as de 6 algoritmos foram utilizadas para validar o método. O mapa gerado utilizando os 30 algoritmos gerou 31 valores de L. Para facilitar os cálculos, os valores que representam os níveis de dificuldade foram normalizados para ajustarem-se no intervalo entre 0 e 1 nos experimentos.

3.2 Cálculo de L utilizando mapas de dificuldade e Seleção do subconjunto representativo

A primeira etapa necessária para selecionar vídeos representativos é a identificar os níveis de dificuldade L de todos os vídeos. A partir dos valores obtidos, escolhe-se um subconjunto que possui potencial de avaliação similar ao do conjunto original. Muitos algoritmos atuais classificam de forma correta os pixels dos quadros da maioria dos vídeos, porém, existem vídeos em que o valor L é bastante elevado. Depois de estimados os valores de L, a etapa final trata da seleção do subconjunto representativo – que possui o mesmo potencial de avaliação do conjunto de vídeos original. A seleção dos vídeos é feita utilizando o Algoritmo 1. Os resultados mostraram que o método proposto selecionou 36 dos 53 vídeos do CDNet 2014. Esses vídeos são listados na Tabela 1, acompanhados dos seus respectivos valores de L.

Tabela 1 – Vídeos representativos selecionados pelo método proposto

Vídeos	L	Vídeos	L	Vídeos	L
tramstop	21,6551	sofa	3,2279	fluidHighway	0,8974
diningRoom	6,0330	winterDriveway	3,0037	blizzard	0,8509
library	18,4753	busyBoulvard	2,0020	canoe	0,8157
parking	5,7412	boats	1,6578	backdoor	0,7564
intermittentPan	5,2249	traffic	1,5795	streetCornerAtNight	0,6812
lakeSide	5,1702	office	1,4662	highway	0,5772
copyMachine	4,9230	winterStreet	1,2747	snowFall	0,4790
fall	4,7769	turbulence0	1,2274	turbulence1	0,3871
zoomInZoomOut	3,9353	twoPositionPTZCam	1,1621	peopleInShade	0,3523
boulevard	3,8582	TunnelExit_0_35fps	1,1102	PETS2006	0,2447
cubicle	3,7995	skating	1,0493	tramCrossroad_1fps	0,1163
corridor	3,4242	wetSnow	0,9910	pedestrians	0,0518

Para comparar os dois conjuntos, foram escolhidos 6 algoritmos. Os desempenhos desses desempenhos foram calculados em dois conjuntos de vídeos (*dataset* original e subconjunto representativo, selecionado pelo método). A Tabela 2 mostra o resultado da comparação dos desempenhos utilizando os dois conjuntos de vídeos. A ordem dos algoritmos considerando seus desempenhos é mantida nas duas avaliações. Além disso, os valores de desempenho (métrica $F1$) de um mesmo algoritmo não apresentam grandes variações nas duas avaliações.

Por meio do método proposto, foi possível reduzir o número total de vídeos (de 53 para 36), mantendo-se o mesmo potencial de avaliação do *dataset*.

Tabela 2 – Desempenhos dos algoritmos utilizando os vídeos originais e o subconjunto selecionado pelo método proposto

Todos os vídeos		**Subconjunto representativo**	
Algoritmo	$F1$	**Algoritmo**	$F1$
FgSegNet-v2	0,9857	FgSegNet-v2	0,9906
BSGAN	0,9350	BSGAN	0,9460
Cascade CNN	0,9200	Cascade CNN	0,9252
DeepBS	0,7467	DeepBS	0,7579
SOBS_CF	0,5944	SOBS_CF	0,5874
Multiscale BG Model	0,5214	Multiscale BG Model	0,5229

4 CONCLUSÕES

Para avaliar o desempenho de algoritmos de detecção de mudanças é importante que o conjunto de vídeos utilizado contenha vídeos com diferentes níveis de dificuldade, para que cada vídeo produza uma informação diferente sobre o desempenho do algoritmo avaliado. Além disso, um conjunto pequeno de vídeos que contenha bom potencial de avaliação pode facilitar a avaliação de versões preliminares de algoritmos. Neste trabalho foram apresentadas abordagens para calcular níveis de dificuldade de vídeos e para selecionar um subconjunto de vídeos representativo de um *dataset*. Os resultados obtidos indicam que o método desenvolvido mostrou-se eficiente uma vez que o subconjunto selecionado apresentou nível de dificuldade similar ao do conjunto original.

AGRADECIMENTOS

Os autores agradecem à Universidade Tecnológica Federal do Paraná e à Fundação Araucária pelo apoio financeiro por meio do Programa de Bolsas de Iniciação Científica e Tecnológica (PIBIC 2020/2021) da acadêmica Katharina Akemi Ikeda Rosa.

REFERÊNCIAS

GOYETTE, N. et al. Changedetection.net: A new change detection benchmark dataset. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. [S.l.: s.n.], jun. 2012. P. 1–8. DOI: https://doi.org/10.1109/CVPRW.2012.6238919.

RUSSEL, J.; COHN, R. **Interquartile Range**. [S.l.]: Tbilisi State University, 2013. ISBN 9785510797251.

SANCHES, Silvio R. R.; OLIVEIRA, Claiton et al. Challenging situations for background subtraction algorithms. **Applied Intelligence**, v. 49, n. 5, p. 1771–1784, mai. 2019. ISSN 1573-7497. DOI: https://doi.org/10.1007/s10489-018-1346-4.

SANCHES, Silvio R. R.; SEMENTILLE, Antonio C. et al. Recommendations for Evaluating the Performance of Background Subtraction Algorithms for Surveillance Systems. **Multimedia Tools and Applications**, Springer, v. 80, n. 3, p. 4421–4454, 2021. DOI: https://doi.org/10.1007/s11042-020-09838-x.

UNIVERSITÉ DE SHERBROOKE. **ChangeDetection.NET – A video database for testing change detection algorithms**. [S.l.: s.n.], 2019. http://www.changedetection.net. Accessed 22 July 2018.