Video Segmentation Learning Using Cascade Residual Convolutional Neural Network

Daniel F. S. Santos†, Rafael G. Pires†
Department of Computing
São Paulo State University
Bauru, Brazil
{danielfssantos1, rafapires}@gmail.com

Danilo Colombo
Cenpes
Petroleo Brasileiro S.A. - Petrobras
Rio de Janeiro - RJ, Brazil
colombo.danilo@petrobras.com.br

João P. Papa
Department of Computing
São Paulo State University
Bauru, Brazil
joao.papa@unesp.br

***Abstract*—Video segmentation consists of a frame-by-frame selection process of meaningful areas related to foreground moving objects. Some applications include traffic monitoring, human tracking, action recognition, efficient video surveillance, and anomaly detection. In these applications, it is not rare to face challenges such as abrupt changes in weather conditions, illumination issues, shadows, subtle dynamic background motions, and also camouflage effects. In this work, we address such shortcomings by proposing a novel deep learning video segmentation approach that incorporates residual information into the foreground detection learning process. The main goal is to provide a method capable of generating an accurate foreground detection given a grayscale video. Experiments conducted on the Change Detection 2014 and on the private dataset PetrobrasROUTES from Petrobras support the effectiveness of the proposed approach concerning some state-of-the-art video segmentation techniques, with overall F-measures of 0.9535 and 0.9636 in the Change Detection 2014 and PetrobrasROUTES datasets, respectively. Such a result places the proposed technique amongst the top 3 state-of-the-art video segmentation methods, besides comprising approximately seven times less parameters than its top one counterpart.**

***Index Terms*—Video Segmentation, Deep Learning, Foreground Object Detection, Residual Map**

I. Introduction

VIDEO segmentation refers to the process of highlighting some specific video image parts that belong to regions of interest, mostly associated to moving objects. Such a task is pretty much complicated to be solved in computer vision, presenting a great number of challenging situations that need to be considered such as extreme weather conditions, camera motion, subtle illumination changes, shadows cast by foreground objects, dynamic background motion, and camouflage. Therefore, addressing these challenges is crucial for the correct functioning of such a variety of computer vision applications including traffic monitoring [1], human tracking [2], action recognition [3], efficient video surveillance [4], and anomaly detection [5].

In the last decades, many non-learning and learning-dependent techniques have been developed to deal with the video segmentation problem. Amongst the non-learning dependent techniques we can highlight simple background subtraction [6], [7], statistical [8]–[10] and fuzzy models [11], subspace learning approaches [12], and robust Principal Component Analysis-based models [13]. Amongst the learning-dependent techniques, we shall cite Quintana and Murguia [14] that proposed a bio-inspired neural system based on Self-organizing Maps and Cellular Neural Networks, called SOM-CNN, to detect dynamic objects in normal and complex scenarios. We can also refer to the work of Schofield et al. [15], that dealt with the problem of people segmentation and counting using a three-stage process: image pre-processing, background identification, and object search. Their method was designed to provide accurate counts, even when the background scene was allowed to vary.

Convolutional Neural Networks (CNNs) have gained quite an attention mostly because of their efficiency in solving tasks involving non-structured data [16], as well as learning translational invariant properties, which is a key point for dealing with background motion detection. Learning-based video segmentation techniques that make use of CNNs are frequently emerging, such as the semi-automatic method for segmenting foreground moving objects proposed by Wang et al. [17], which consists in two main objectives: (i) to produce segmentation maps sufficiently accurate to be used as a ground truth, and (ii) to avoid, as much as possible, user interventions. Another example concerns some state-of-the-art video segmentation techniques such as FgSegNet_S and FgSegNet_M [18], which are characterized mainly for being robust deep convolutional autoencoder networks that can be trained in an end-to-end model using a few video frames.

Some years ago, the concept of “residual learning" arose to highlight the importance of considering skip connections to avoid a variety of deep network problems, such as vanishing gradients and overfitting. In this paper, we proposed a robust deep learning video segmentation technique that consists in a cascade CNN model that incorporates residual information [19]–[21] into the learning process for foreground object detection. To the best of our knowledge, such an approach has never been investigated in the video segmentation domain. Experiments conducted in the public Change Detection 2014 (CD2014) and in the private PetrobrasROUTES datasets support the effectiveness of the proposed approach when detecting changes in indoor and outdoor camera-captured videos.

†These authors contributed equally to this paper.

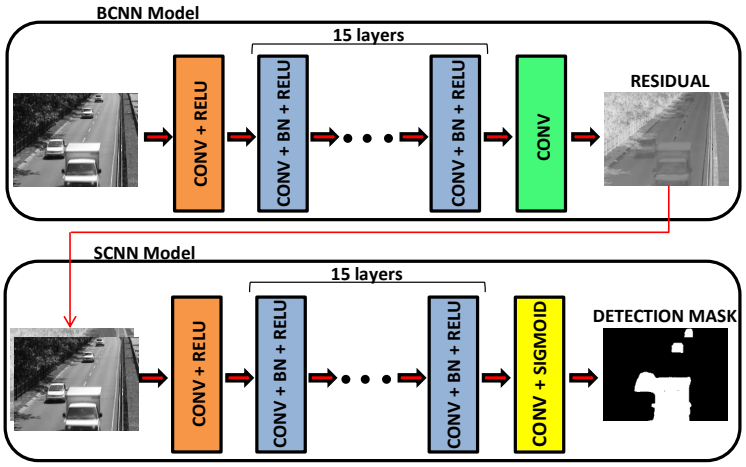

Fig. 1. Architecture of the proposed CRCNN approach.

II. Proposed Approach

In this work, we proposed a novel approach named "Cascade-Residual Convolutional Neural Network" (CRCNN) for video segmentation purposes, which was highly influenced by the works of Zhang et al. [19] concerning non-blind denoising using residual learning, and Wang et al. [17] regarding segmentation issues. Besides, we also considered concepts from the work of Zhang et al. [22] with respect to blind denoising learning. Figure 1 depicts the proposed approach.

The deep learning video segmentation pipeline adopted in this paper is the same one proposed by Lim and Keles [18], and consists in three main steps: (i) to annotate foreground objects in a small subset of frames collected from the video of interest, (ii) to train the CRCNN model in a supervised fashion, and (iii) to further apply the trained model over each image frame extracted from the video of interest to generate its correspondent binary foreground detection mask.

The proposed CRCNN uses a two-stage video processing strategy. In the first step, given a grayscale version of the video image frame to be segmented, a first deep residual CNN, hereinafter called "Background Convolutional Neural Network" (BCNN), is used to generate the correspondent residual map. Further, this output combined with its residual map is presented to a second CNN, named "Segmentation Convolution Neural Network" (SCNN), which is then used to generate the foreground detection mask. The next sections describe more details about the proposed CRCNN model.

A. Background Convolution Neural Network

The proposed Background Convolutional Neural Network learns how to infer, given an input grayscale video frame, which parts do not correspond to the background areas. To create a robust background image, we used the approach proposed by Bevilacqua [23] to allow BCNN modeling the input non-background information in the form of a residual map.

The BCNN training step consists of two phases: the first one takes an interval $I = \{\beta_1, \beta_2, ..., \beta_n\}$ of consecutive frames from the video and uses it to calculate the *deterministic background image*, which stands for an image b that represents the median of such an interval. The second phase consists of minimizing the mean square error between the deterministic background image and the *approximated background image* a, which is represented as follows:

$$a = \sigma(f - BCNN(f; \Theta_1)), \tag{1}$$

where $\sigma(\cdot)$ stands for the logistic-sigmoid function, f denotes the input image normalized between $[0, 1]$, Θ_1 refers to the BCNN trainable parameters, and $BCNN(\cdot;\cdot)$ refers to the residual map learned during the training process. In light of that, the BCNN training process aims at minimizing the following equation:

$$L_B(b, f; \Theta_1) = \frac{1}{2m} \sum_{i=1}^{m} ||b_i - a_i||_F^2, \tag{2}$$

where m stands for the number of training samples and $||\cdot||_F^2$ represents the Frobenius norm. Notice that we employed a patch-based methodology, where b_i and a_i denote the i^{th} patch extracted from images b and a, respectively.

B. Segmentation Convolutional Neural Network

The Segmentation Convolutional Neural Network learns how to detect foreground objects present in the video frames. For such purpose, it uses the information provided by the image frames and also their residual maps. The detections are presented in the form of binary images, in which white pixels correspond to the foreground object locations.

The SCNN training process differs from the BCNN one in basically two points: (i) the network input, that is composed of a concatenation between the grayscale image frame and its residual map counterpart (i.e., the output generated as a result of forward propagating the grayscale image through the trained BCNN), and (ii) the training process, which aims at minimizing the average binary cross-entropy measured

between the network output and the ground-truth binary detection mask. Such an image corresponds to the pre-annotated true foreground objects present in the grayscale input image. Therefore, the SCNN training process aims at minimizing the following equation:

$$L_S(g, c; \Theta_2) = -\frac{1}{m}\sum_{i=1}^{m}[g_i \log(\hat{g}_i) + (1-g_i)\log(1-\hat{g}_i)], \quad (3)$$

where

$$\hat{g} = SCNN(c; \Theta_2). \quad (4)$$

Notice that g is the ground-truth pre-annotated binary mask, Θ_2 stands for the SCNN trainable parameters and c indicates the SCNN depth concatenation input between f and its residual map. Besides, g_i and $\hat{g}_i$ denote the i^{th} patch extracted from images g and $\hat{g}$, respectively.

C. Cascade Residual Convolutional Neural Network

As depicted in Figure 1, the proposed Cascade Residual Convolutional Neural Network is composed of two main models, i.e., the BCNN and SCNN, which are connected by the residual map generated by the former network. Table I presents a summary of the CRCNN configuration parameters, where the dimensions of the convolution kernels are represented by three-dimensional vectors. The first and second dimensions represent the kernel width and height, respectively, and the third dimension denotes the number of outputs that will be generated after the convolution step.

TABLE I
CRCNN ARCHITECTURE SPECIFICATION. THE TABLE USES THE SAME COLOR CODES AS IN FIGURE 1 TO REPRESENT THE DIFFERENT CRCNN LAYERS.

	Orange	**Blue**	**Green**	**Yellow**
Kernel szs.	3 x 3 x 64	3 x 3 x 64	3 x 3 x 1	3 x 3 x 1
Activation	ReLU	ReLU	Linear	Sigmoid
Batch norm.	No	Yes	No	No

Three different kinds of activations were used, i.e., a linear function (applied to the convolution layers only), a rectified linear unit (ReLU), and a sigmoid function. Batch normalization [24] was also applied in all 15 layers, placed at the middle of the BCNN and SCNN models, i.e., before the application of the ReLU activation function.

III. METHODOLOGY

In this section, we present the methodology used to train and evaluate the proposed CRCNN model. For the sake of clarification, we divided the section into three parts: (i) Section III-A presents all the relevant information about the datasets used in this work, (ii) Section III-B details the CRCNN training process, and (iii) Section III-C discusses the detection and evaluation procedures.

A. Datasets

1) Change Detection Dataset 2014: The Change Detection Dataset 2014 (CD2014) is a large and freely available dataset of videos collected from different realistic, camera-captured, and challenging scenarios [25]. Such a dataset contains 11 video categories with 4 to 6 video sequences each, as presented in Table II.

TABLE II
CD2014 DATASET SPECIFICATION.

Category	**Qnt. Videos**	**Qnt. Frames**
Baseline	4	6,049
Dynamic Background	6	18,871
Camera Jitter	4	6,420
Intermitt. Obj. Motion	6	18,650
Shadow	6	16,949
Thermal	5	21,100
Bad Weather	4	20,900
Low Framerate	4	9,400
Night Videos	6	16,609
PTZ	4	8,630
Turbulence	4	15,700
Total	**53**	**159,278**

The CD2014 categories include:

- **Baseline**: combines mild challenges present in Dynamic Background, Camera Jitter, Intermittent Object Motion, and Shadow categories. It serves mainly as a starting point to adjust the segmentation technique.
- **Dynamic Background**: includes scenes with so much background motion, e.g., cars and trucks passing in front of a tree shaken.
- **Camera Jitter**: contains indoor and outdoor videos captured by unstable video devices, for example vibrating cameras.
- **Intermittent Object Motion**: contains objects that move and then stop for a short while producing “ghosting" artefacts.
- **Shadow**: indoor and outdoor videos containing objects surrounding by a strong shadow that could be miss detected as real moving objects.
- **Thermal**: videos that have been captured by far-infrared cameras.
- **Bad Weather**: includes outdoor videos captured from challenging winter weather conditions, e.g., snow storms, and fog.
- **Low Framerate**: videos captured varying frame-rates between 0.17fps and 1fps.
- **PTZ**: videos captured by pan-tilt-zoom cameras.
- **Turbulence**: outdoor videos that show air turbulence caused by rising heat.

2) PetrobrasROUTES: The PetrobrasROUTES is a private dataset which consists of 281 high-resolution color images collected from an indoor Petrobras[1] workspace. The main challenge of such dataset regards the detection of objects obstructing escape routes.

B. Training procedure

To train the proposed CRCNN model over CD2014 dataset, we employed the following protocol:

1) to select 300 color images and their 300 correspondent binary images, which were ground-truth manually annotated.
2) to convert the 300 color images to their grayscale versions and use the first 100 images to calculate the deterministic background.
3) to normalize the remaining 200 grayscale images[2] by subtracting them the average grayscale value.
4) to subdivide the images into small patches using 50% to 75% of overlap depending on the image height and width dimensions[3].
5) to subdivide the deterministic background image into small patches and then replicate them so that every input grayscale patch has its deterministic background grayscale patch counterpart.

We employed the following protocol to train the proposed CRCNN model over PetrobrasROUTES dataset:

1) to select 51 color images and their 51 correspondent binary images, which were ground-truth manually annotated.
2) to convert the 51 color images to their grayscale versions and use one of them as the deterministic background.
3) to normalize the remaining 50 images by subtracting them the average grayscale value.
4) to subdivide the training and deterministic background images following same steps 4) and 5) from CD2014 dataset training protocol.

The BCNN and SCNN models were trained using the Adam method [28] by a maximum of 50 epochs[4] using a learning rate[5] of 0.001 and batches of size 128. We trained the BCNN and SCNN models with 80% of the patches, and used the remaining 20% to evaluate the convergence of the training process.

[1]Petrobras is a publicly-held company on an integrated basis and specialized in the oil, natural gas, and energy industry [26].

[2]We used the same set of training images from [27] to train the proposed CRCNN model.

[3]The sizes and the overlapping rates were empirically defined taking into account the usage of larger overlaps for small images and dimensions of the patches limited to 50 pixels (i.e., for both height and width). Besides avoid much slowness during the training process, the usage of patches works like a natural data augmentation that prevents deep learning issues such as overfitting.

[4]Depending on the training process convergence the maximum epoch value can be less than 50.

[5]The initial value is reduced by a factor of 0.1 every time the loss function hits a plateau.

C. Evaluation procedure

The evaluation process consists in to apply the trained CRCNN model over each video test image as follows:

- **Deep Segmentation**: such a step consists in first forward propagating the test images through the trained BCNN model, and further using the residual input (i.e., the input grayscale image and its residual counterpart) to feed the trained SCNN. Later, we binarized[6] the SCNN probabilistic output.
- **Misclassification Rate**: in such a step, we calculated the number of correct and incorrect detections encoded by the True Positives (TPs), i..e, the number of pixels correctly classified as foreground, the True Negatives (TNs), i.e., the number of pixels correctly classified as background, the False Positives (FPs), i.e., the number of background pixels incorrectly classified as foreground, and the False Negatives (FNs), i.e., the number of foreground pixels incorrectly classified as background.
- **Detection Measurements**: in such a step, the classification rates are combined into four different measures that provide a more clever way to measure the robustness of the proposed CRCNN model. Those measures are computed as follows:

$$Precision = \frac{TP}{TP + FP}, \tag{5}$$

$$Recall = \frac{TP}{TP + FN}, \tag{6}$$

$$F - measure = 2.0 \times \frac{Recall \times Precision}{Recall + Precision}, \tag{7}$$

and

$$PWC = 100.0 \times \frac{FN + FP}{TP + FP + FN + TN} \tag{8}$$

where PWC denotes the percentage of wrong classifications.

IV. Experimental Section

In this section, we present the experimental results regarding the methodology described earlier considering each dataset.

A. Results over CD2014 Dataset

Table III presents the overall and per-category F-measure values. One can observe the proposed CRCNN model overcomes the supervised learning methods Cascade [17] and DeepBS [29], being also more accurate in comparison to the non-learning-based techniques, i.e., SuBSENSE [30], IUTIS-5 [31], and PAWCS [32]. According to Table III, the proposed technique achieved results that are pretty much close to the

[6]In the majority of the experiments, the best threshold value was set to 0.8, but in some rare cases, it has been set to 0.6.

TABLE III
A COMPARISON OF F-MEASURE RESULTS OF 11 CATEGORIES FROM CD2014 DATASET

Methods	Baseline	C.Jitter	B.Waet	Dyn.Bg.	Int.Obj.	L.Frame	N.Videos	PanTZ	Shadow	Thermal	Turbul.	Overall
FgSegNet_S	**0.9980**	**0.9951**	**0.9902**	**0.9902**	**0.9942**	**0.9511**	**0.9837**	**0.9837**	**0.9967**	**0.9945**	**0.9796**	**0.9878**
FgSegNet_M	**0.9975**	**0.9945**	**0.9838**	**0.9838**	**0.9933**	**0.9558**	**0.9779**	**0.9779**	**0.9954**	**0.9923**	**0.9776**	**0.9865**
CRCNN	**0.9919**	**0.9799**	**0.9569**	**0.9687**	**0.9755**	0.8498	**0.9388**	**0.8967**	**0.9852**	**0.9818**	**0.9637**	**0.9535**
Cascade	0.9786	0.9758	0.9451	0.9451	0.8505	**0.8804**	0.8926	0.8926	0.9593	0.8958	0.9215	0.9272
DeepBS	0.9580	0.8990	0.8647	0.8647	0.6097	0.5900	0.6359	0.6359	0.9304	0.7583	0.8993	0.7593
IUTIS-5	0.9567	0.8332	0.8289	0.8289	0.7296	0.7911	0.5132	0.5132	0.9084	0.8303	0.8507	0.7820
PAWCS	0.9397	0.8137	0.8059	0.8059	0.7764	0.6433	0.4171	0.4171	0.8934	0.8324	0.7667	0.7477
SuBSENSE	0.9503	0.8152	0.8594	0.8594	0.6569	0.6594	0.4918	0.4918	0.8986	0.8171	0.8423	0.7453

state-of-the-art ones, as one can notice in the categories “Baseline", “Camera Jitter", and “Shadow", where CRCNN results are quite similar to the FgSegNet_S and FgSegNet_M [18] techniques, with F-measure differences of only 0.01 (approximately). Notice the proposed approach comprises $1,112,770$ parameters, which turns out to be a more compact architecture with respect to FgSegNet_S, which has $7,622,465$ parameters.

Table IV highlights the robustness of the proposed CRCNN model, placing it as the third best approach concerning the measures used in this work, only behind FgSegNet_S and FgSegNet_M models. Also, the precision results differ from FgSegNet_S and FgSegNet_M by around 0.02, while recall values differ by around 0.03.

TABLE IV
COMPARISON OF PRECISION, RECALL AND PWC OVERALL RESULTS FROM CD2014 DATASET.

Methods	Avg. Precision	Avg. Recall	Avg. PWC
FgSegNet_S	**0.9751**	**0.9896**	**0.0461**
FgSegNet_M	**0.9758**	**0.9836**	**0.0559**
CRCNN	**0.9604**	**0.9602**	**0.1348**
Cascade	0.8997	0.9506	0.4052
DeepBS	0.8332	0.7545	1.9920
IUTIS-5	0.8087	0.7849	1.1986
PAWCS	0.7857	0.7718	1.1992
SuBSENSE	0.7509	0.8124	1.6780

Additionally to the results presented in Tables III and IV, Figure 2 depicts three foreground detection masks, each one from a different category. Notice the gray-tone areas presented in Figure 2c stand for regions that do not belong to the region of interest. From the ground-truth presented in Figure 2c, one can observe the proposed CRCNN model produced a more accurate and precise detection binary masks for the “Shadow" category (Figure 2d). These results are better than the ones obtained by FgSegNet_S, Cascade, and DeepBS techniques. Concerning the “Thermal category", CRCNN outperformed Cascade and DeepBS models.

One can also observe that CRCNN exhibited false negative detections to a greater extent when compared to FgSegNet_S and Cascade techniques in the “Night Videos" category. However, such a category poses a challenge to all compared methods either, since Cascade and FgSegNet_S results exhibit false positive detections both, and DeepBS was not capable of detecting the moving cars in the video frame. A closer look at the second row from Figure 2 evidenced that BCNN model smoothed areas corresponding to foreground regions during its learning process. We hypothesized that such regions are used by SCNN during its learning process as a clue indicating which locations are the most probable to encode scene changes.

B. Results over PetrobrasROUTES Dataset

Table V presents the overall results comparing FgSegNet_S with the proposed technique. One can observe the proposed CRCNN achieved the best results in almost all measures, with differences of around 0.1 and 0.2 in terms of recall and PWC, respectively.

TABLE V
COMPARISON OF PRECISION, RECALL AND PWC OVERALL RESULTS FROM PETROBRASROUTES DATASET.

Avg. Measures	FgSegNet_S	CRCNN
F-measure	0.9221	**0.9619**
Precision	**0.9770**	0.9611
Recall	0.8732	**0.9627**
PWC	0.4287	**0.2218**

Additionally to the results presented in Table V, Figure 3a depicts a video frame of a scape route containing an undesirable object. Regards its ground-truth (Figure 3b), one can observe the proposed CRCNN has been more accurate than FgSegNet_S (Figures 3c and 3d), with the detection results limited only to the object central region.

V. Conclusions

In this work, we proposed a novel cascade convolutional neural network which uses a residual learning strategy in an attempt to solve video segmentation problems. Regarding CD2014 dataset, the proposed CRCNN model achieved results close to the state-of-the-art ones, which were obtained by FgSegNet_S and FgSegNet_M techniques. Besides, the method was capable of overcoming two supervised learning methods and three other non-supervised segmentation techniques in terms of F-measure, precision, recall, and PWC overall results. Concerning PetrobrasROUTES dataset, the proposed CRCNN model outperformed the state-of-the-art FgSegNet_S method in terms of F-measure, recall, and PWC overall results. Besides, we state that better results can be possibly achieved by fine-tuning the patch sizes.

Regarding future works, we pretend to investigate the CRCNN behavior under such a fine-tuning process carefully, and also search for other possible ways to apply the residual learning in video segmentation tasks. As a starting point, we intend to investigate the usage of color images and possibly other CNN architecture configurations.

Acknowledgment

The authors are grateful to CNPq grants 307066/2017-7 and 427968/2018-6, FAPESP grants 2013/07375-0, 2014/12236-1 and 2016/19403-6, as well as Petrobras grant 2017/00285-6.

References

[1] J. Kato, T. Watanabe, S. Joga, J. Rittscher, and A. Blake, "An HMM-based segmentation method for traffic monitoring movies," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1291–1296, 2002.

[2] J. Zhou and J. Hoang, "Real time robust human detection and tracking system," in *Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2005, pp. 149–149.

[3] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2012.

[4] S. Brutzer, B. Höferlin, and G. Heidemann, "Evaluation of background subtraction techniques for video surveillance," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 1937–1944.

[5] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *Computing Surveys*, vol. 41, no. 3, p. 15, 2009.

[6] D. Sakkos, H. Liu, J. Han, and L. Shao, "End-to-end video background subtraction with 3d convolutional neural networks," *Multimedia Tools and Applications*, vol. 77, no. 17, pp. 23 023–23 041, 2018.

[7] M. Braham and M. Van Droogenbroeck, "Deep background subtraction with scene-specific convolutional neural networks," in *International Conference on Systems, Signals and Image Processing*. IEEE, 2016, pp. 1–4.

[8] A. Lanza and L. Di Stefano, "Statistical change detection by the pool adjacent violators algorithm," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1894–1910, 2011.

[9] S. Varadarajan, P. Miller, and H. Zhou, "Spatial mixture of gaussians for dynamic background modelling," in *International Conference on Advanced Video and Signal Based Surveillance*. IEEE, 2013, pp. 63–68.

[10] J. D. Pulgarin-Giraldo, A. Alvarez-Meza, D. Insuasti-Ceballos, T. Bouwmans, and G. Castellanos-Dominguez, "GMM background modeling using divergence-based weight updating," in *Iberoamerican Congress on Pattern Recognition*. Springer, 2016, pp. 282–290.

[11] T. Bouwmans, "Background subtraction for visual surveillance: A fuzzy approach," *Handbook on soft computing for video surveillance*, vol. 5, pp. 103–138, 2012.

[12] D. Farcas, C. Marghes, and T. Bouwmans, "Background subtraction via incremental maximum margin criterion: a discriminative subspace approach," *Machine Vision and Applications*, vol. 23, no. 6, pp. 1083–1101, 2012.

[13] S. Javed, T. Bouwmans, and S. K. Jung, "Combining ARF and OR-PCA for robust background subtraction of noisy videos," in *International Conference on Image Analysis and Processing*. Springer, 2015, pp. 340–351.

[14] J. A. Ramirez-Quintana and M. I. Chacon-Murguia, "Self-adaptive SOM-CNN neural system for dynamic object detection in normal and complex scenarios," *Pattern Recognition*, vol. 48, no. 4, pp. 1137–1149, 2015.

[15] A. Schofield, P. Mehta, and T. J. Stonham, "A system for counting people in video images using neural networks to identify the background scene," *Pattern Recognition*, vol. 29, no. 8, pp. 1421–1428, 1996.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[17] Y. Wang, Z. Luo, and P.-M. Jodoin, "Interactive deep learning method for segmenting moving objects," *Pattern Recognition Letters*, vol. 96, pp. 66–75, 2017.

[18] L. A. Lim and H. Y. Keles, "Foreground segmentation using convolutional neural networks for multiscale feature encoding," *Pattern Recognition Letters*, vol. 112, pp. 256–262, 2018.

[19] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising," *Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Conference on Computer vision and Pattern Recognition*, 2016, pp. 770–778.

[21] J. Pang, W. Sun, J. S. Ren, C. Yang, and Q. Yan, "Cascade residual learning: A two-stage convolutional neural network for stereo matching," in *International Conference on Computer Vision*, 2017, pp. 887–895.

[22] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a fast and flexible solution for CNN-based image denoising," *Transactions on Image Processing*, vol. 27, no. 9, pp. 4608–4622, 2018.

[23] A. Bevilacqua, L. Di Stefano, and A. Lanza, "A simple self-calibration method to infer a non-parametric model of the imaging system noise," in *Workshops on Applications of Computer Vision*, vol. 1. IEEE, 2005, pp. 229–234.

[24] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[25] "Changedetection.net (CDNET)," http://changedetection.net, accessed: 2019-05-29.

[26] "Petrobras," http://www.petrobras.com.br/en, accessed: 2019-06-04.

[27] "Foreground segmentation network version 2," https://github.com/lim-anggun/FgSegNet_v2, accessed: 2019-05-29.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[29] M. Babaee, D. T. Dinh, and G. Rigoll, "A deep convolutional neural network for video sequence background subtraction," *Pattern Recognition*, vol. 76, pp. 635–649, 2018.

[30] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, "Subsense: A universal change detection method with local adaptive sensitivity," *Transactions on Image Processing*, vol. 24, no. 1, pp. 359–373, 2014.

[31] S. Bianco, G. Ciocca, and R. Schettini, "Combination of video change detection algorithms by genetic programming," *Transactions on Evolutionary Computation*, vol. 21, no. 6, pp. 914–928, 2017.

[32] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, "A self-adjusting approach to change detection based on background word consensus," in *Winter Conference on Applications of Computer Vision*. IEEE, 2015, pp. 990–997.

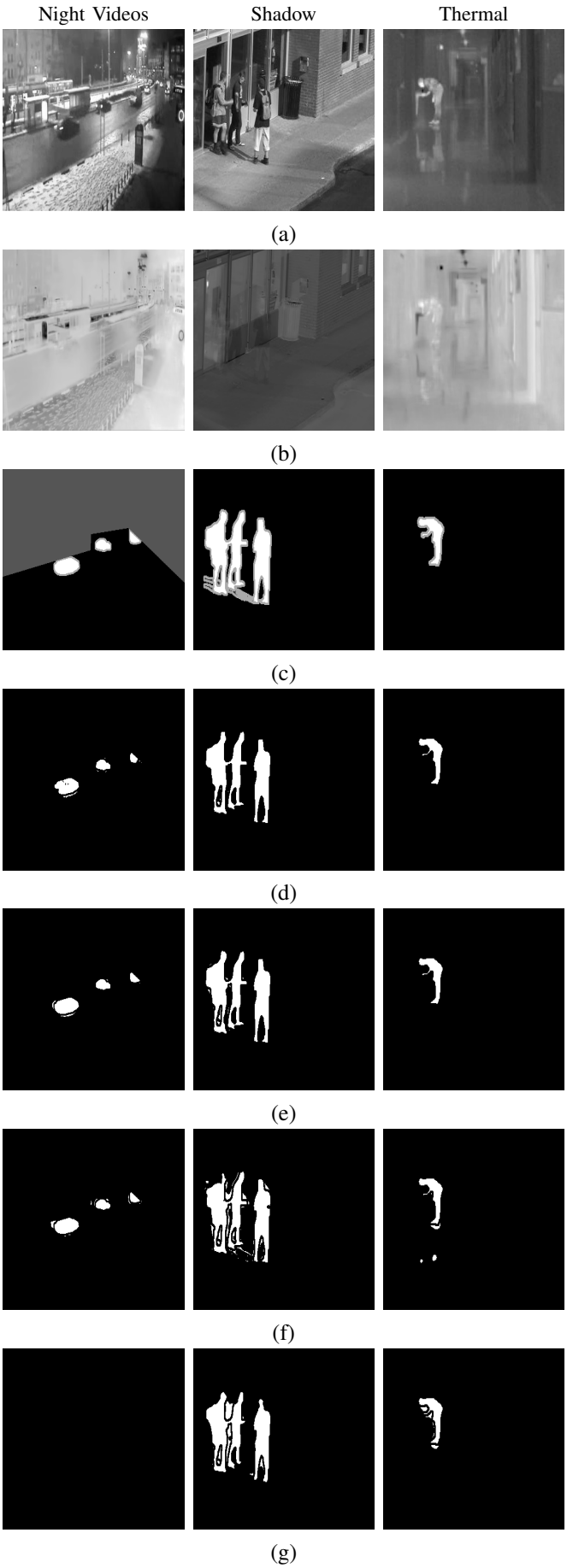

Fig. 2. Qualitative results considering the categories “Night Videos", “Shadow", and “Thermal" from CD2014 dataset: (a) input grayscale frame, (b) residual maps, (c) ground-truth detection masks, results concerning (d) proposed CRCNN, (e) FgSegNet_S, (f) Cascade, and (g) DeepBS models.

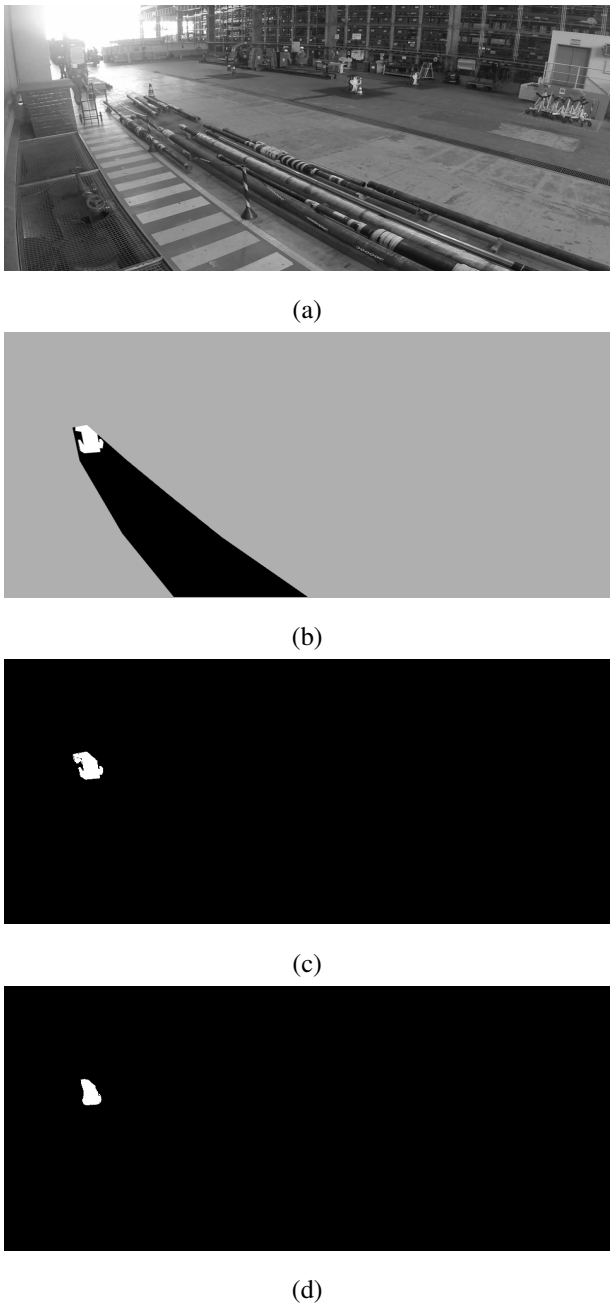

Fig. 3. Qualitative results considering an obstructed route video scene from PetrobrasROUTES dataset: (a) input grayscale frame, (b) ground-truth detection mask, and results concerning (c) CRCNN and (d) FgSegNet_S techniques.