Early Wildfire Smoke Detection Based on Motion-Based Geometric Image Transformation and Deep Convolutional Generative Adversarial Networks

2019 IEEE International Conference on Acoustics, Speech and Signal Processing

Süleyman Aslan, Uğur Güdükbay, B. Uğur Töreyin, A. Enis Çetin

May 17, 2019

Introduction

- Immediate and accurate detection of wildfire is essentially important in forest monitoring systems
  - One of the most harmful hazards in rural areas
- For wildfire detection, the use of visible-range video captured by surveillance cameras are suitable
  - They can be deployed and operated in a cost-effective manner [1]
- The challenge is to provide a robust detection system with negligible false positive rates
- If the flames are visible, they can be detected by analyzing the motion and color clues of a video [2]
  - Modeling various spatio-temporal features and dynamic texture analysis have been shown to be able to detect fire as well [3]

1. A. E. Çetin, K. Dimitropoulos, B. Gouverneur, N. Grammalidis, O. Gunay, Y. H. Habibolu, B. U. Töreyin, and S. Verstockt, "Video fire detection–review," Digital Signal Processing, vol. 23, no. 6, pp. 1827–1843, 2013.
2. Y. Dedeoglu, B. U. Toreyin, U. Güdükbay, and A. E. Cetin, "Real-time fire and flame detection in video," in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing. IEEE, 2005, vol. 2 of ICASSO'05, pp. ii–669
3. K. Dimitropoulos, P. Barmpoutis, and N. Grammalidis, "Spatio-temporal flame modeling and dynamic texture analysis for automatic video-based fire detection," IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, no. 2, pp. 339–351, 2015.

Introduction

- Focus on smoke detection
  - Smoke rises above the crowns of trees, and it has a higher chance of falling into the viewing range of cameras

- Particularly challenging because of false alarms due to cloud shadows and fog [4]

- Deep Convolutional Generative Adversarial Networks (DCGANs) [5] based wildfire smoke detection
  - DCGANs can learn representations of regular wilderness images in an unsupervised manner

4. Y. Zhao, J. Ma, X. Li, and J. Zhang, "Saliency detection and deep learning-based wildfire identification in UAV imagery," Sensors, vol. 18, no. 3, pp. Article No. 712, 19 pages, 2012.
5. A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," CoRR, vol. abs/1511.06434, 2015

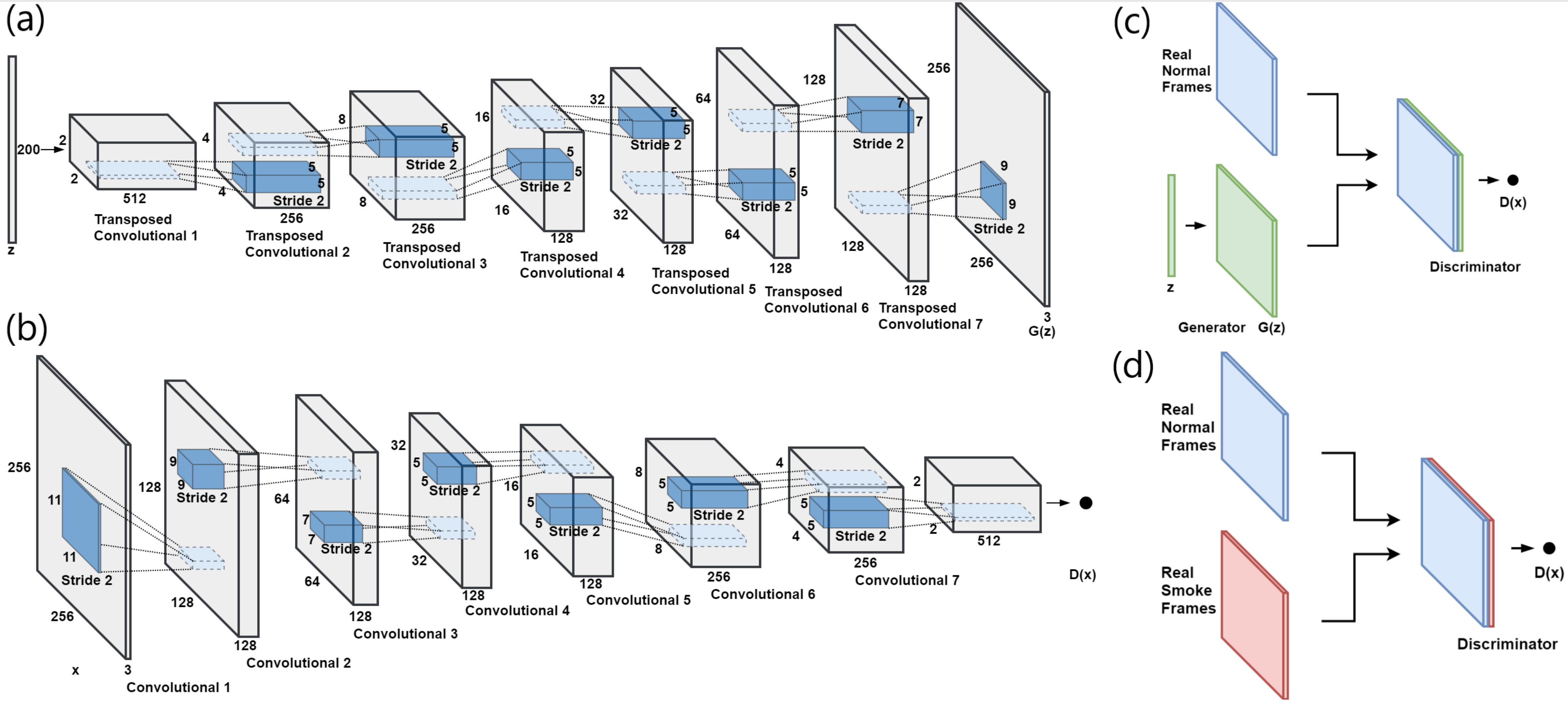
Introduction

- The contributions are,
- Using discriminator network of the DCGAN as a classifier that distinguishes ordinary image sequences without smoke from wildfire smoke
- A two-stage training approach for a DCGAN
  - Unsupervised representation learning in combination with supervised classification learning
- Exploiting the wildfire smoke's evolution in time
  - Temporal progress of smoke is integrated by using a motion-based image transformation before training

Method

- Method is based on a DCGAN structure accepting images with size 256×256 px
- Seven transposed convolutional layers for the generator, and seven convolutional layers for the discriminator
- In the first stage, we train DCGAN using images without smoke and noise distribution z
  - Discriminator network learns a representation for ordinary wilderness video scenes and distinguishes smoke
- Then, we refine and retrain the discriminator without generator network in the second stage
  - regular video images obtained from the cameras constitute the “real” training data
  - actual smoke images correspond to generated data
- Adversarial training makes the recognition system more robust and the second stage of training increases the recognition accuracy

Method



The architecture of DCGAN: (a) generator network, (b) discriminator network, (c) the first stage of training, and (d) the second stage of training.

Method

- Instance normalization [6] in the discriminator network, and batch normalization [7] in the generator network
- “MSRA” initialization [8] to initialize the layers and dropout layers [9]
- Adam optimizer for stochastic optimization [10]
- Algorithms are supported by TensorFlow [11]

6. D. Ulyanov, A. Vedaldi, and V. S. Lempitsky, “Instance normalization: The missing ingredient for fast stylization,” CoRR, vol. abs/1607.08022, 2016..
7. S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” CoRR, vol. abs/1502.03167, 2015
8. K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification,” in Proceedings of the IEEE International Conference on Computer Vision, 2015, ICCV’15, pp. 1026–1034
9. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” The Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929– 1958, 2014
10. D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” CoRR, vol. abs/1412.6980, 20147
11. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., “Tensorflow: a system for large-scale machine learning,” in Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation, 2016, OSDI’16, pp. 265–283

Method

- Transformations to the frames as a pre-processing step
  - Exploiting the temporal behavior of smoke

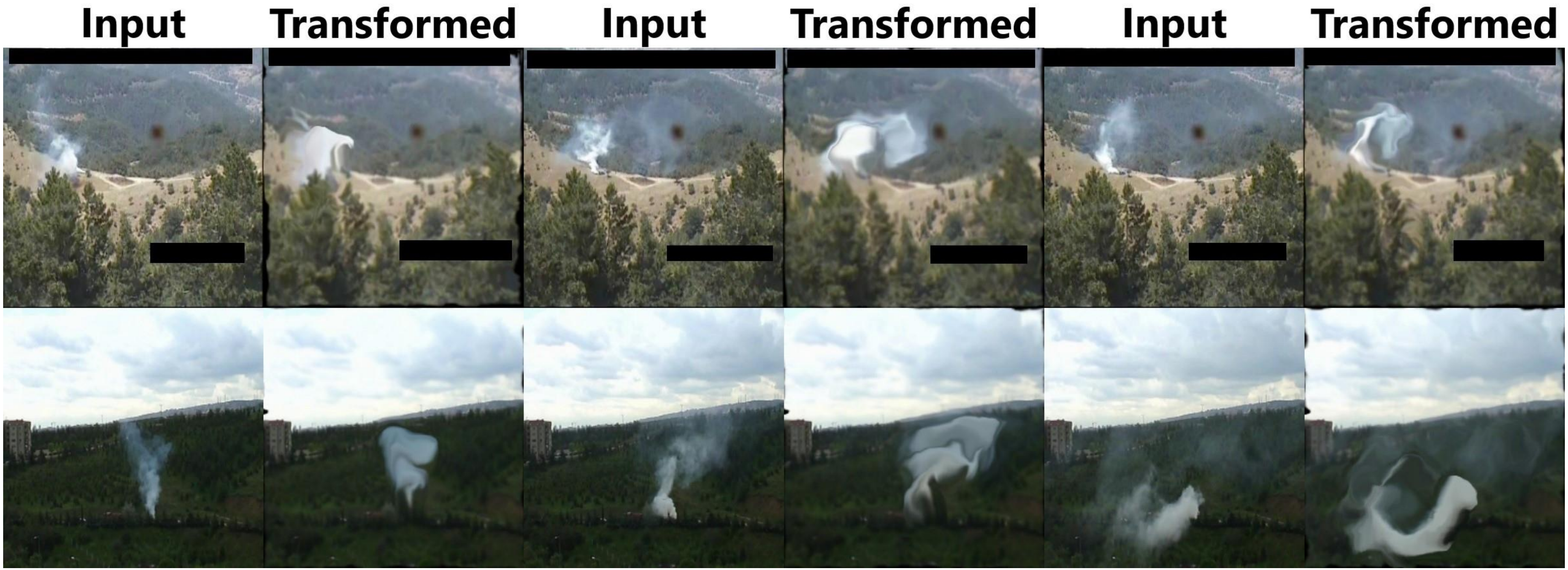
- Estimated motion is computed using Farnebacks algorithm [12], then a geometrical transformation is applied as follows

$$T(k,l) = S(k - f_k(k,l), l - f_l(k,l))$$

- where,
  - $T(k,l)$ is the pixel at position (k,l) in the resulting transformed image
  - $S(k,l)$ is the pixel at position (k,l) in the source image
  - $f_k(k,l)$ is the estimated motion along horizontal axis at position (k,l)
  - $f_l(k,l)$ is the estimated motion along vertical axis at position (k,l)

12. G. Farneback, "Two-frame motion estimation based ¨ on polynomial expansion," in Proceedings of the 13th Scandinavian Conference on Image Analysis, Berlin, Heidelberg, 2003, SCIA'03, pp. 363–370, SpringerVerlag

Method



Examples of transformed frames.

Method

- Wildfire smoke has no particular shape, therefore we treat smoke as an unusual event or an anomaly

- In standard GAN training, the discriminator D is updated using the stochastic gradient

$$SG_1 = \nabla_{\theta_d} \frac{1}{M} \sum_{i=1}^{M} (\log D(x_i) + \log(1 - D(G(z_i))))$$

- where,
    - x_i and z_i are the i-th regular image data and noise vector, respectively
    - G represents the generator that generates a "fake" image according to the input noise vector z_i
    - the vector θ_d contains the parameters of the discriminator

- Generator network G is "adversarially" trained
    - Smoke videos are not included in this stage

Method

- This GAN is able to detect smoke, because smoke images are not in the training set

- To increase the recognition accuracy, we perform a second round of training by fine-tuning the discriminator using the stochastic gradient

$$SG_2 = \nabla_{\theta_d} \frac{1}{L} \sum_{i=1}^{L} (\log D(x_i) + \log(1 - D(y_i))$$

- where y_i represents the i-th image containing wildfire smoke

- Number of smoke image samples, L, is much smaller than the size of the initial training set, M, containing regular forest and wilderness images

- Generator network of GAN is not used in this stage

Experimental Results

- 40 video clips containing no smoke frames and 29 video clips containing only smoke frames

- For each smoke video, there is a corresponding “normal” video for generator network to learn
  - However, not all normal videos have a corresponding smoke video

- We first apply motion-based geometrical image transformation and normalize the duration of videos

- Data is split the into training, validation, and test sets with a ratio of 3:1:1
  - Some smoke videos are completely removed from validation and training sets
  - However, all normal videos are included in the training set

Experimental Results

- We first evaluate the proposed method in terms of frame-based results
- We compare our model by excluding the contributions one by one
- Our approach targets at reducing the false positive rate, while keeping the hit-rate as high as possible
- Best results on the test set

Method	**TNR (%)**	**TPR (%)**
Our method	99.45	86.23
Transformation excluded	98.70	83.33
Refinement excluded	95.10	62.56
Transformation and refinement excluded	93.94	60.16
Adversarial training excluded	98.07	84.10
Adversarial training and transformation excluded	97.39	81.43

Obtained true negative rate (TNR) and true positive rate (TPR) values on test set for frame-based evaluation.

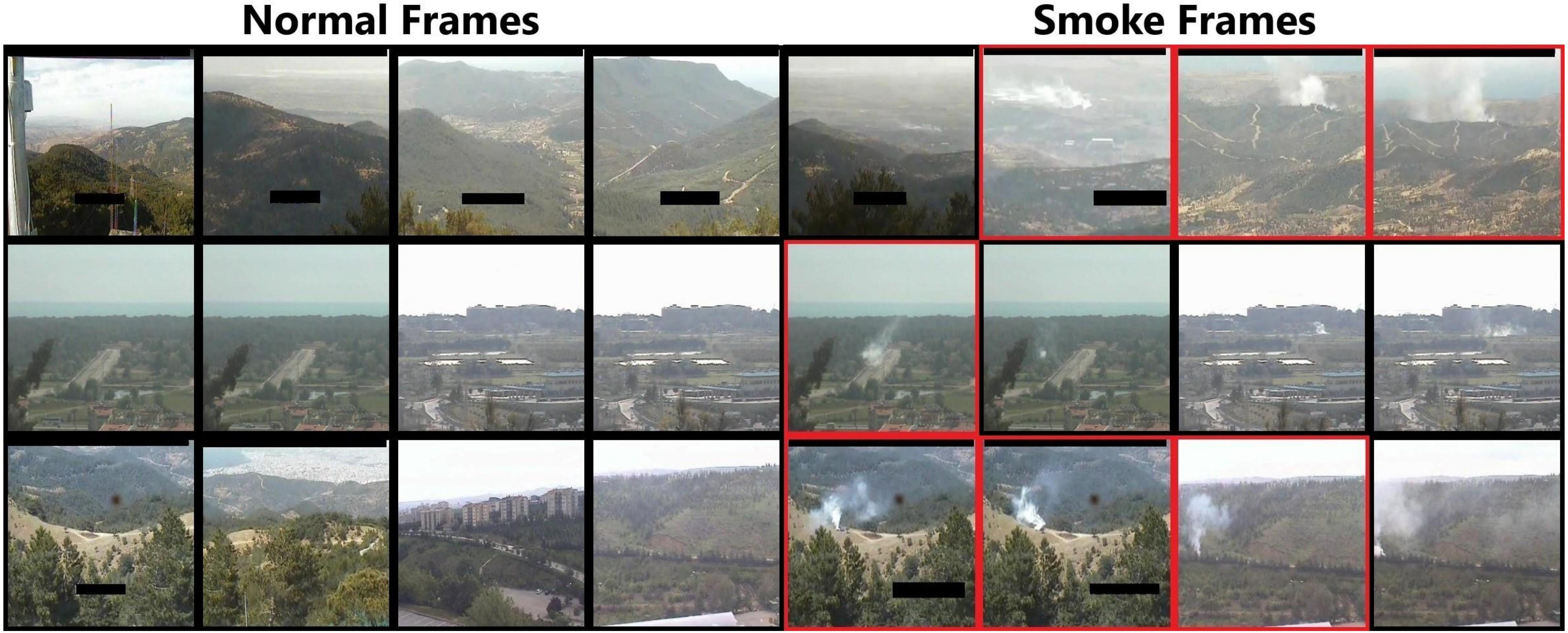
Experimental Results

- For the motion-based transformation, the difference is mainly in hit-rates
- Without the refinement stage, detection rates are smaller
  - However it can still be useful when there are no labeled smoke frames
- Without adversarial training, the model is more susceptible to false positives

Method	**TNR (%)**	**TPR (%)**
Our method	99.45	86.23
Transformation excluded	98.70	83.33
Refinement excluded	95.10	62.56
Transformation and refinement excluded	93.94	60.16
Adversarial training excluded	98.07	84.10
Adversarial training and transformation excluded	97.39	81.43

Obtained true negative rate (TNR) and true positive rate (TPR) values on test set for frame-based evaluation.

Experimental Results



Examples of frame-based classification results. Red border indicates that smoke is detected in that frame.

Experimental Results

- We also evaluate the approach in terms of video-based results
  - A video is classified as a smoke video, if, at least, one frame is detected as smoke
- We train different versions of the network, by up-weighting and down-weighting the cost of a false positive relative to a false negative
- False-positive rate of 2.5% is achieved corresponding to a 6.9% miss rate

	TNR (%)	**TPR (%)**
Up-weighted false positives	100.00	89.67
Unweighted	97.50	93.10
Down-weighted false positives	87.50	100.00

Video-based results for our method.

Conclusion

- A wildfire smoke detection method using DCGANs and motion-based geometrical image transformation
- Treating smoke as an unusual event
- Two-stage DCGAN training approach
- Spatio-temporal dynamics of smoke are acquired using motion-based geometric image transformation
- Results suggest that the proposed method achieves low false alarm rates while keeping the detection rate high
  - It can also be utilized to detect other anomalous events in forests, such as, flames or people in restricted zones

References

1. A. E. Çetin, K. Dimitropoulos, B. Gouverneur, N. Grammalidis, O. Gunay, Y. H. Habibolu, B. U. Töreyin, and S. Verstockt, “Video fire detection–review,” Digital Signal Processing, vol. 23, no. 6, pp. 1827–1843, 2013.

2. Y. Dedeoglu, B. U. Toreyin, U. Güdükbay, and A. E. Cetin, “Real-time fire and flame detection in video,” in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing. IEEE, 2005, vol. 2 of ICASSO’05, pp. ii–669

3. K. Dimitropoulos, P. Barmpoutis, and N. Grammalidis, “Spatio-temporal flame modeling and dynamic texture analysis for automatic video-based fire detection,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, no. 2, pp. 339–351, 2015.

4. Y. Zhao, J. Ma, X. Li, and J. Zhang, “Saliency detection and deep learning-based wildfire identification in UAV imagery,” Sensors, vol. 18, no. 3, pp. Article No. 712, 19 pages, 2012.

5. A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” CoRR, vol. abs/1511.06434, 2015

6. D. Ulyanov, A. Vedaldi, and V. S. Lempitsky, “Instance normalization: The missing ingredient for fast stylization,” CoRR, vol. abs/1607.08022, 2016..

7. S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” CoRR, vol. abs/1502.03167, 2015

8. K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification,” in Proceedings of the IEEE International Conference on Computer Vision, 2015, ICCV’15, pp. 1026–1034

9. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” The Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929– 1958, 2014

10. D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” CoRR, vol. abs/1412.6980, 20147

11. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., “Tensorflow: a system for large-scale machine learning,” in Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation, 2016, OSDI’16, pp. 265–283

12. G. Farneback, “Two-frame motion estimation based ¨ on polynomial expansion,” in Proceedings of the 13th Scandinavian Conference on Image Analysis, Berlin, Heidelberg, 2003, SCIA’03, pp. 363–370, SpringerVerlag