Paper Review

1. Unmasking the abnormal events in video
2. Anomaly Detection in Video Sequence with Appearance-Motion Correspondence

1. Unmasking the abnormal events in video

- Author:

  Radu Tudor Ionescu, Sorina Smeureanu, Bogdan Alexe, Marius Popescu
- Source

  ICCV-2017

Abstract

- Contributions:
  - Completely unsupervised strategy
  - First work to apply *unmasking*
    - *‘A technique is based on testing the degradation rate of the cross-validation accuracy of learned models, as the best features, are iteratively dropped from the learning process.’*
  - Running in real-time at 20 FPS

Video Anomaly Detection

- Problem recall:
  - Depends very much on context
  - Impossible to find all kinds of anomalies
  - Both appearance and motion information are important
- General approach
  - Learn a model of normality, then detect outlier as anomaly.
  - Employe deep features

Method Overview

- 8-step pipline:
  - Step A - C: frames labelling
  - Step D: features extracting
  - Step E-G: unmasking
  - Step H: abnormality assigning

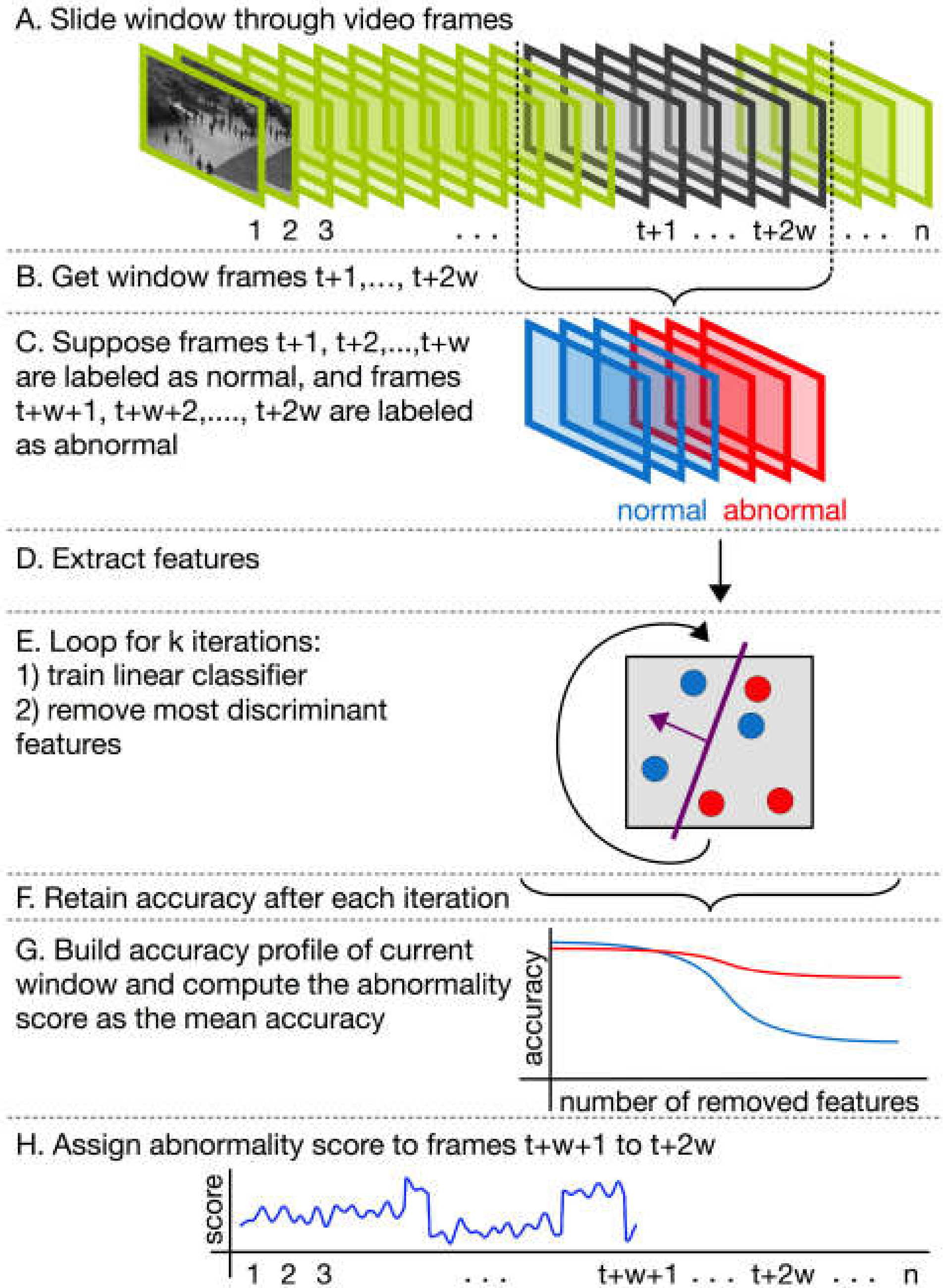
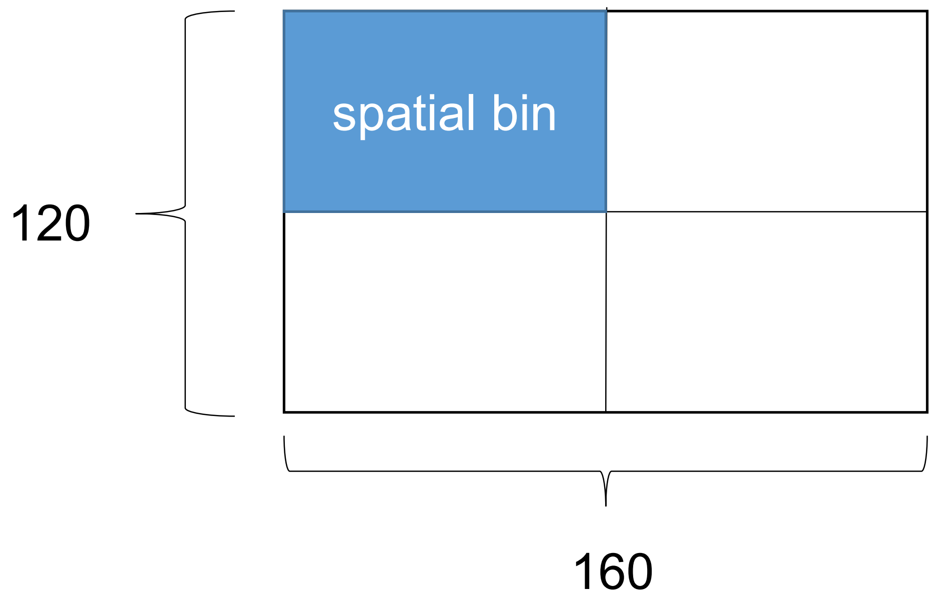


Figure 1. Our anomaly detection framework based on unmasking [12]. The steps are processed in sequential order from (A) to (H). Best viewed in color.

Data preprocessing

- Divide frames into 2*2 bins as follows:

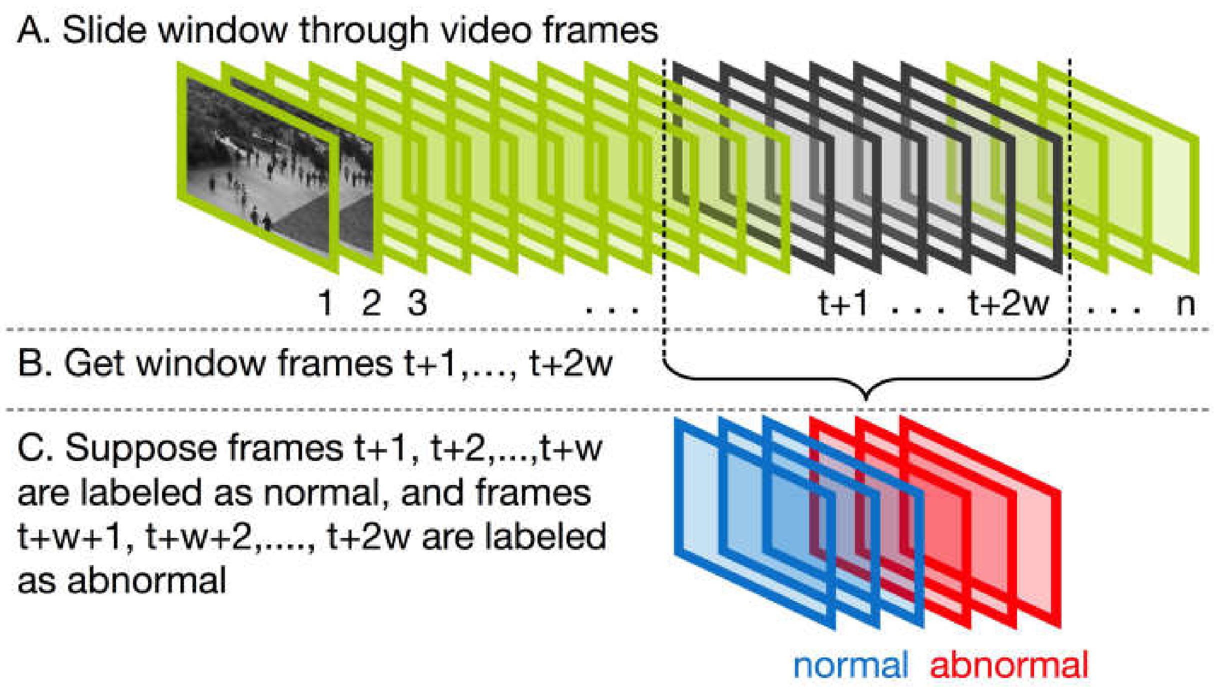


- Process each bin individually until Step G

Step A-C: frames labelling

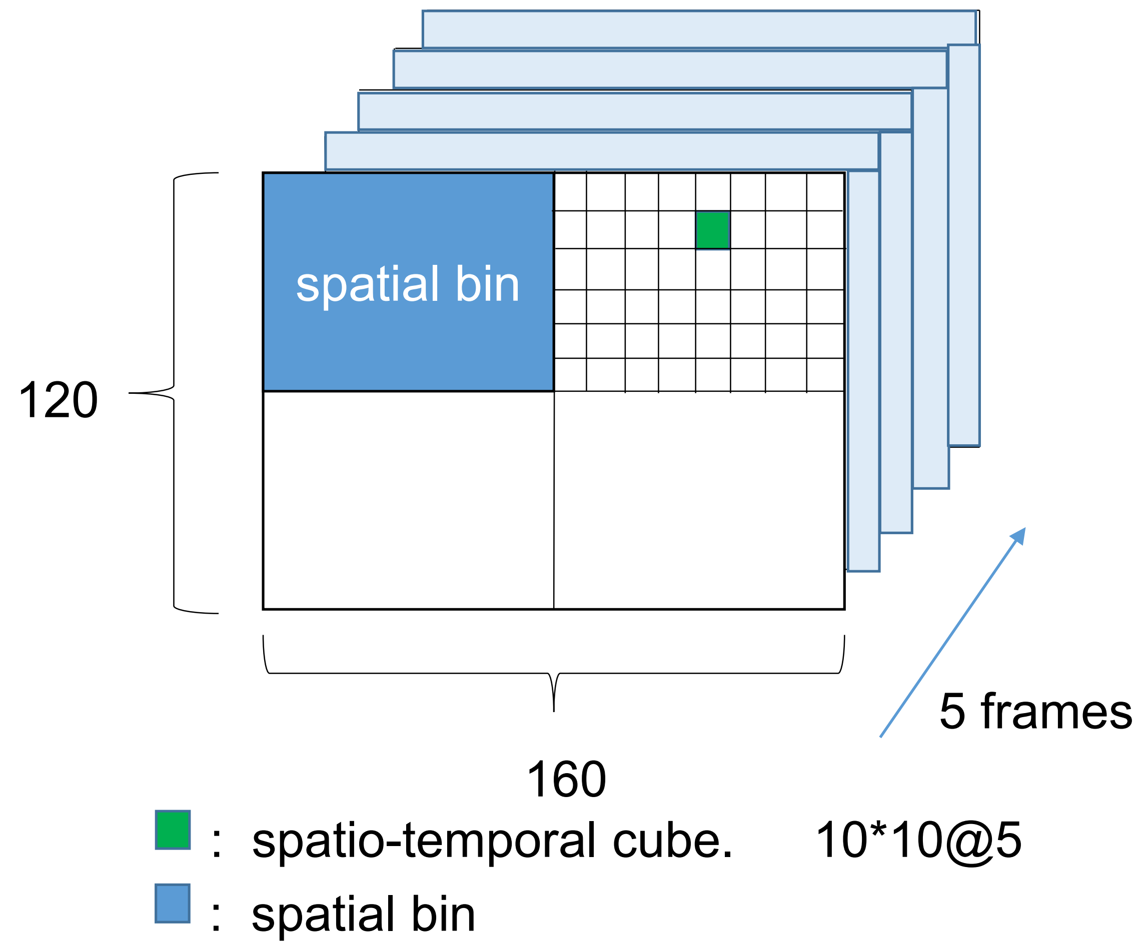
- At first, we suppose the left half as normal, the right half as abnormal.

- Then, we seek to find if this hypothesis is true indeed.



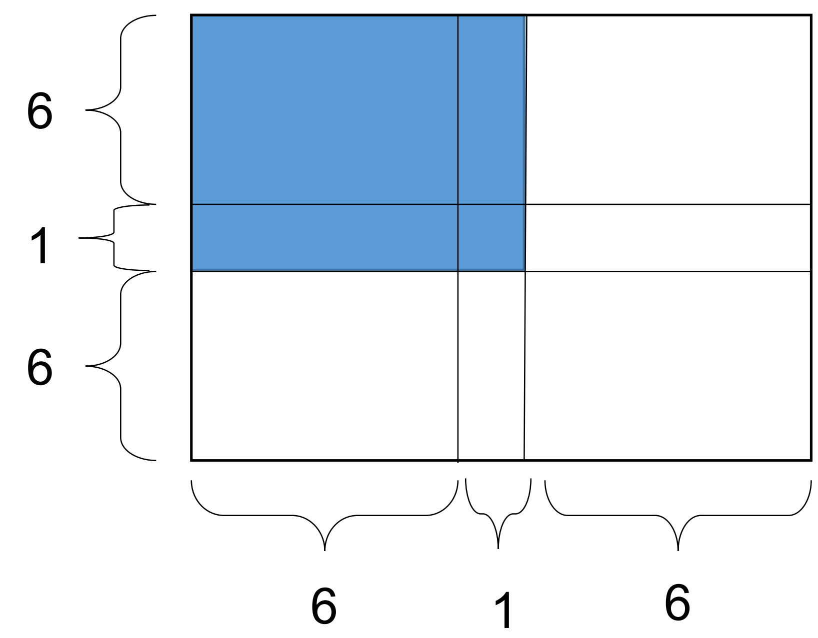
Step D: feature extracting

- Motion features
  - Compute 3D gradient motion features (500 dimensions) for each cube.
  - Eliminate the cubes without motion gradient (stay static actually).
  - Each cube is treated as an example in step E.



Step D: feature extracting

- Appearance features
  - Use Conv5 of VGG-f.
  - Reshape 7*7@256 into 12544 (7*7*256) component.

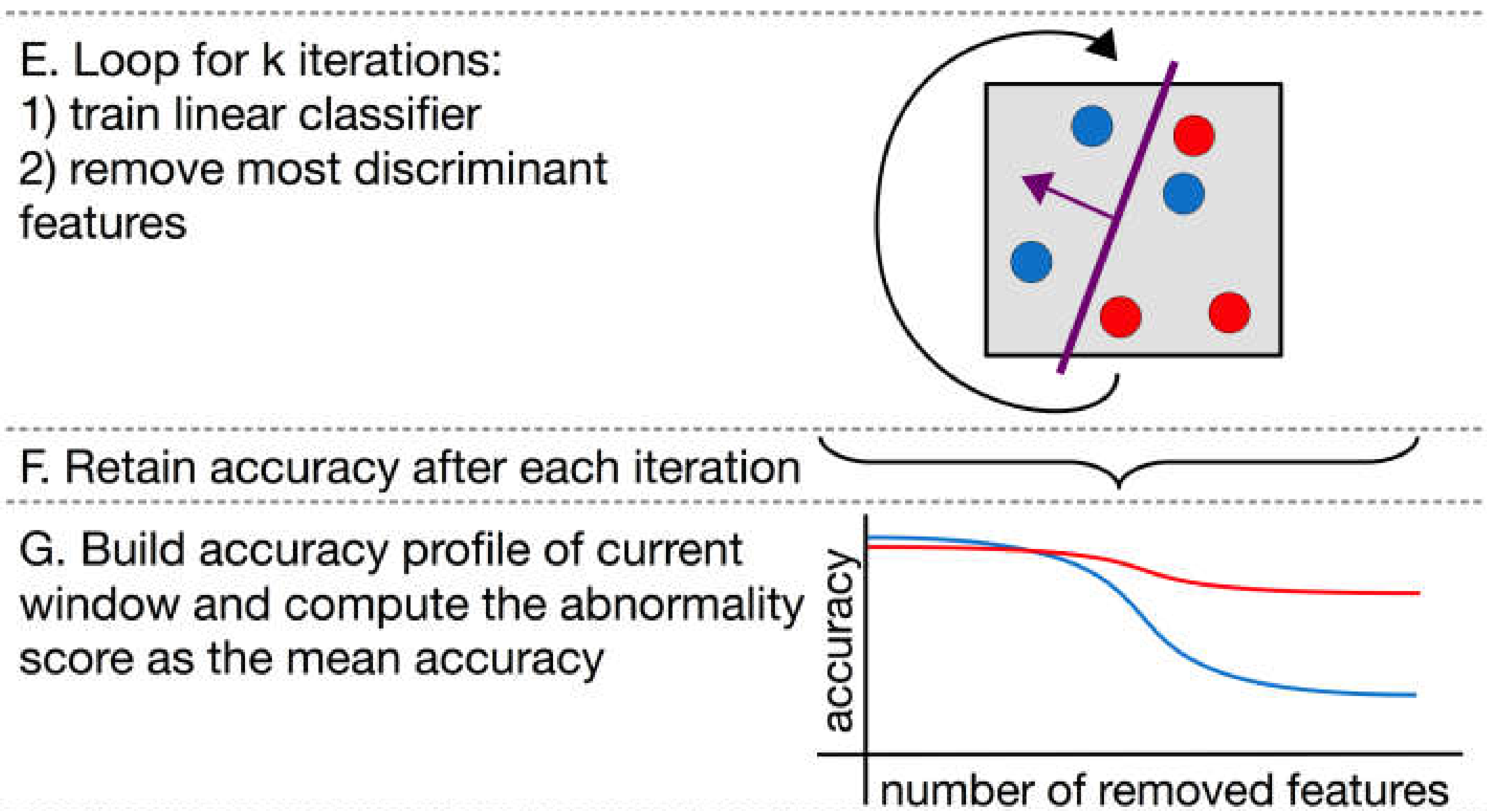


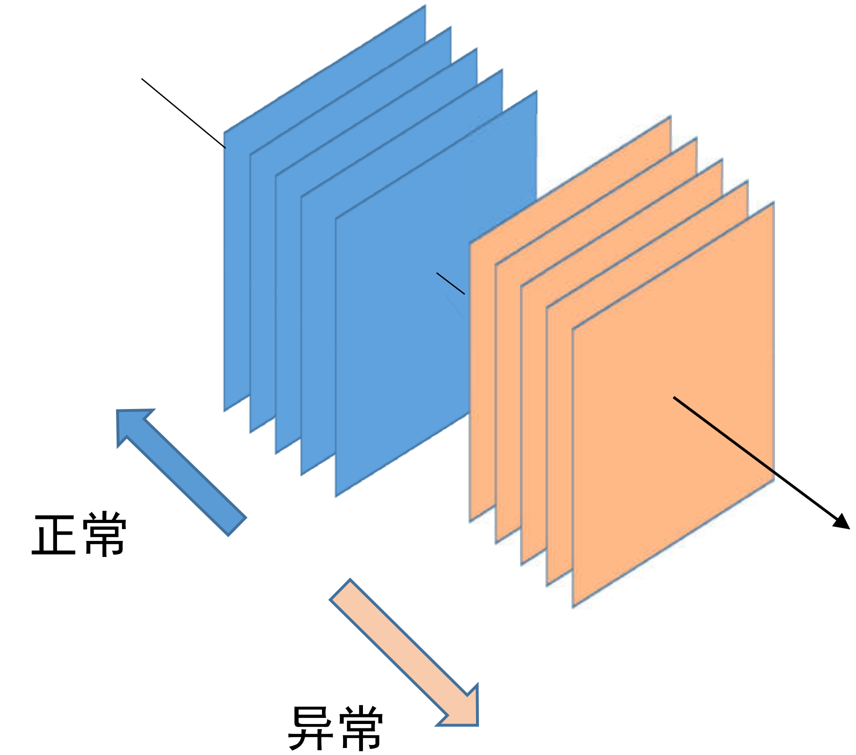
Conv5 feature maps: 13*13@256

Step E-G: unmasking

- Some differences from [6]:
  - Using traning accuracy instead of CV-Acc.
  - The size of “sliding window” is 2*w.
- Hypothesis:
  - If two consecutive events are both normal, whatever differences threr are, only a small number of features will be reflected.
  - If the accuracy of the Logistic Regression Classifier drop down suddenly, we treat the last half frames as normal at that time.

Step E-G: unmasking



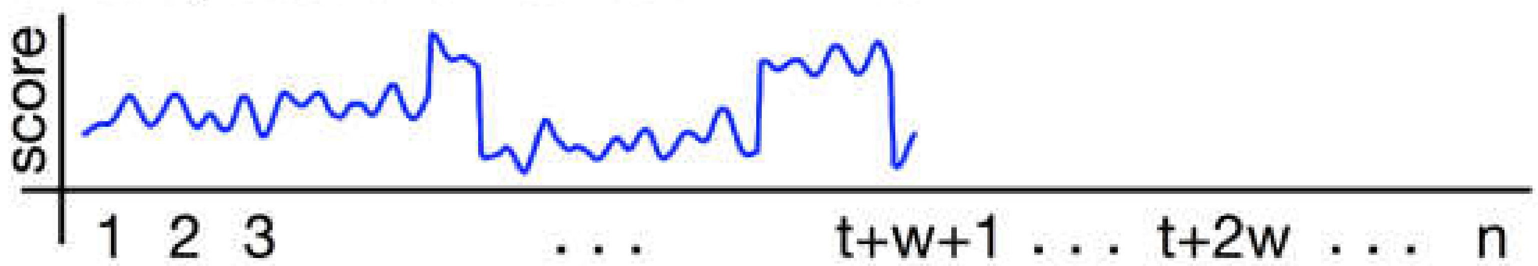


- 分类器越精确，说明二者越容易区分，即特征差别越明显，判别后者为异常。
- 分类器越不精确，说明二者越不容易区分，即特征几乎没差别，判别后者为正常。

Set the average of the traning acc over k loops as the anomaly score of the last w frames.

Step E-G: abnormality assigning

- Move 2*w window at stride s:
  - E.g. s=1, w=10, the abnormality of a specific frame is obtained by averaging the anomaly scores which obtained after processing evry separate window that includes the respective frame in its second half.
- Assign abnormality to each frame:
  - H. Assign abnormality score to frames t+w+1 to t+2w



2. Anomaly Detection in Video Sequence with Appearance-Motion Correspondence

- Author

  Trong-Nguyen Nguyen, Jean Meunier

- Source

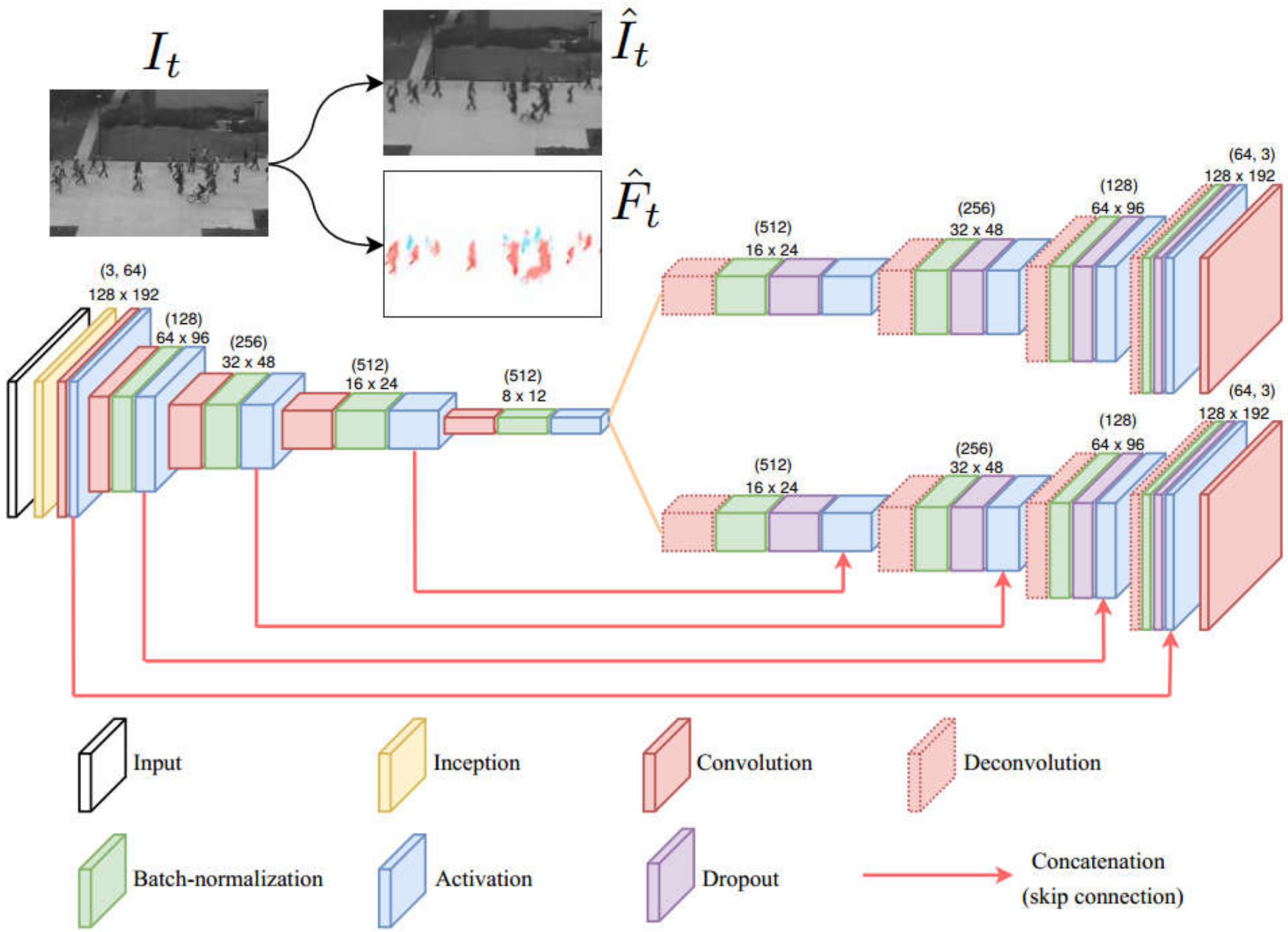
  ICCV-2019

Abstract

- Contributions:
  - A combination of U-Net and Conv-AE
  - Integrate an Inception module
  - Propose a ”patch-based scheme” to estimate frame-level normality

Method Overview

- Features:
  - Add Inception module right after input layer.
  - Conv-AE only for appearance.
  - U-Net for motion prediction.



Dive a bit deeper

- Inception module
  - Apply this to let model select appropriate Conv Ops.
- Conv-AE

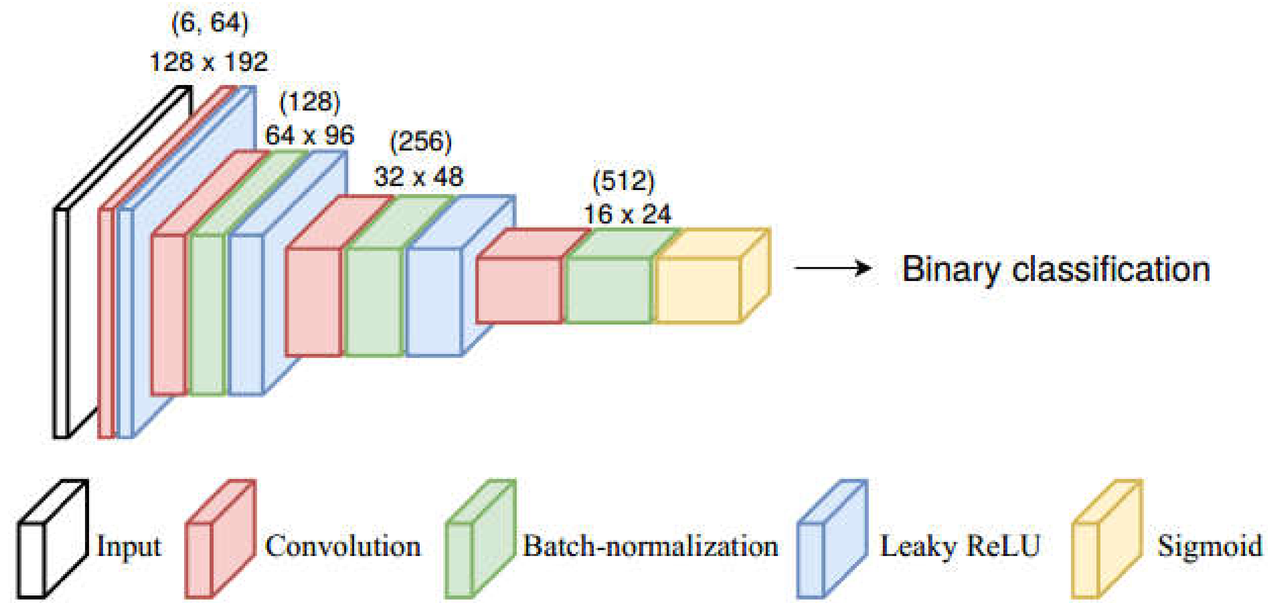
$$\mathcal{L}_{int}(I, \hat{I}) = ||I - \hat{I}||_2^2$$

$$\mathcal{L}_{grad}(I, \hat{I}) = \sum_{d \in \{x,y\}} \left\| |g_d(I)| - |g_d(\hat{I})| \right\|_1$$

$$\mathcal{L}(I, \hat{I}) = \mathcal{L}_{int}(I, \hat{I}) + \mathcal{L}_{grad}(I, \hat{I})$$

- U-Net
  - Focus on learning the association between appearance patterns and corresponding motions. (人和车对应有不同的动作)

Dive a bit deeper (co



- U-Net
  - Focus on learning the association between appearance patterns and corresponding motions. (人和车对应有不同的动作)
  - Use FlowNet2 [15] to estimate ground truth Optical Flow.
  - Motion loss $\mathcal{L}_{flow}(F_t, \hat{F}_t) = ||F_t, \hat{F}_t||_1$
- GAN loss
  - $\mathcal{L}_{\mathcal{D}}(I, F, \hat{F}) = \frac{1}{2}\sum_{x,y,c} -\log \mathcal{D}(I, F)_{x,y,c} + \frac{1}{2}\sum_{x,y,c} -\log\left[1 - \mathcal{D}(I, \hat{F})_{x,y,c}\right]$
  - $\mathcal{L}_{\mathcal{G}}(I, \hat{I}, F, \hat{F}) = \lambda_{\mathcal{G}}\sum_{x,y,c} -\log \mathcal{D}(I, \hat{F})_{x,y,c} + \lambda_a \mathcal{L}_{\text{appe}}(I, \hat{I}) + \lambda_f \mathcal{L}_{\text{flow}}(F, \hat{F})$

Anomaly Detection

- Patch-based score esitimation scheme
  - P indicates an image patch (set to 16*16 in reported experiments)

$$\begin{cases} \mathcal{S}_I(P) = \frac{1}{|P|}\sum_{i,j\in P}\left(I_{i,j} - \hat{I}_{i,j}\right)^2 \\ \mathcal{S}_F(P) = \frac{1}{|P|}\sum_{i,j\in P}\left(F_{i,j} - \hat{F}_{i,j}\right)^2 \end{cases}$$

$$\mathcal{S} = \log[w_F \mathcal{S}_F(\tilde{P})] + \lambda_S \log[w_I \mathcal{S}_I(\tilde{P})]$$

$$\tilde{P} \leftarrow \underset{P \text{ slides on frame}}{\operatorname{argmax}} \mathcal{S}_F(P)$$

  - Weight parameters estimating strategy

$$\begin{cases} w_F = \left[\frac{1}{n}\sum_{i=1}^{n}\mathcal{S}_{F_i}\left(\tilde{P}_i\right)\right]^{-1} \\ w_I = \left[\frac{1}{n}\sum_{i=1}^{n}\mathcal{S}_{I_i}\left(\tilde{P}_i\right)\right]^{-1} \end{cases}$$

- 概括S中的log部分的含义：

1. 当前帧的差距和训练数据统计得到的平均最大差距之比
2. 当前差距越大，log越大，S越大，越有可能是异常帧

Anomaly Detection (cont'd)

- Patch-based score esitimation scheme
  - Score normalization $\hat{\mathcal{S}}_t = \frac{\mathcal{S}_t}{\max(\mathcal{S}_{1..m})}$