There’s no difference: Convolutional Neural Networks for transient detection without template subtraction

Tatiana Acero-Cuellar, Federica Bianco, Gregory Dobler, Masao Sako, Helen Qu

17 November 2022

Content

- ➔ **Motivation**
- ➔ **Models Description (CNN, Data)**
- ➔ **Models Result**
- ➔ **Feature Importance Analysis**

Main Purpose

➔ **Convolutional Neural Network** without difference image. (Difference Image Analysis most computationally expensive step in transient detection).

➔ **Reduce** computation cost of transient detection.

‘partially avoid the DIA process’

Similar works

Effective image differencing with convolutional neural networks for real-time transient hunting

Nima Sedaghat[1★] and Ashish Mahabal[2★]

[1]*Department of Computer Science, University of Freiburg, Georges-Koehler-Allee 052, D-79110 Freiburg, Germany*
[2]*Center for Data Driven Discovery, Caltech, 1200 E California Blvd, Pasadena, CA 91125, USA*

Deep Learning for Image Sequence Classification of Astronomical Events

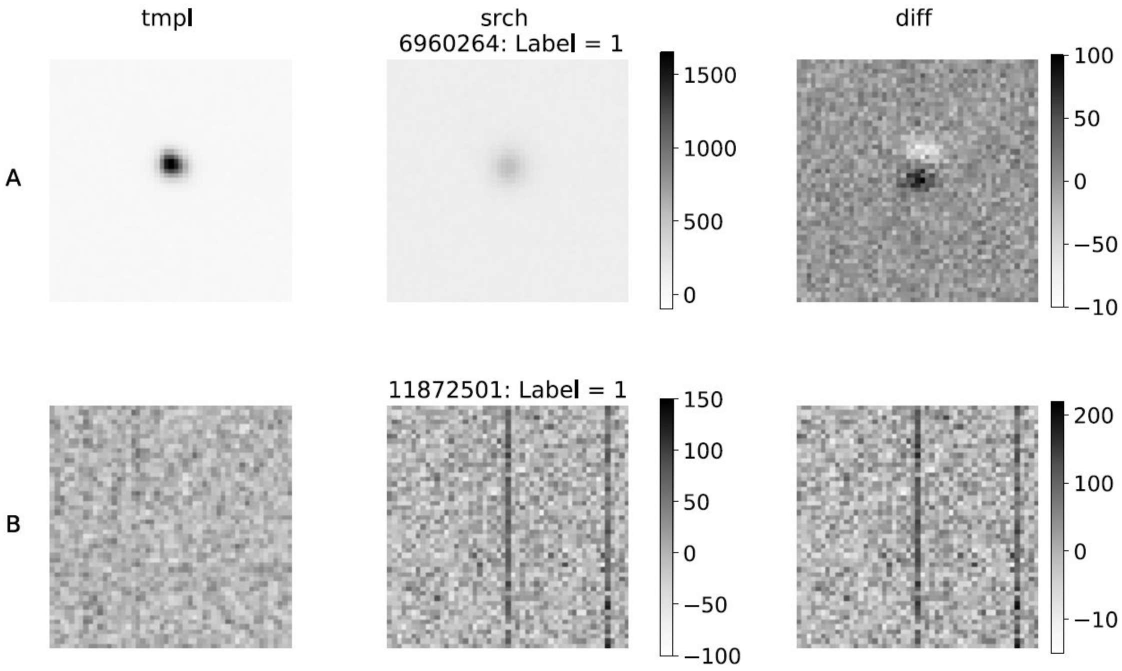
Rodrigo Carrasco-Davis[1,7], Guillermo Cabrera-Vives[2,7], Francisco Förster[6,7], Pablo A. Estévez[1,7], Pablo Huijse[3,7], Pavlos Protopapas[5], Ignacio Reyes[1,7], Jorge Martínez-Palomera[4,6,7], and Cristóbal Donoso[2]

Detecting optical transients using artificial neural networks and reference images from different surveys

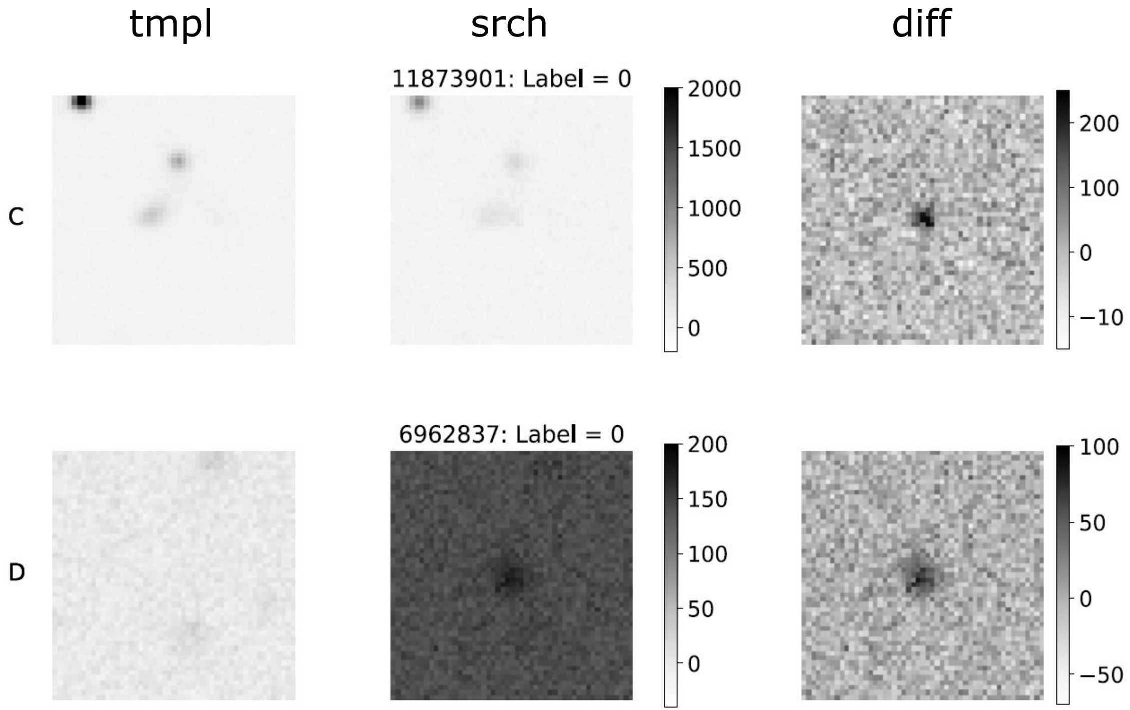
Katarzyna Wardęga,[1,2★] Adam Zadrożny,[3†] Martin Beroiz,[1] Richard Camuccio [1,4] and Mario C. Díaz[1]

Similar works done by Sedaghat & Mahabal (2018), Carrasco-Davis et al.(2019) and Wardęga et al. (2020)

Example of “Bogus” type



Example of “Real” type



Data

➔ The data used in this work consists of postage stamps of images collected by the DES during its first observational season (Y1), August 2013 through February 2014.

➔ The data corresponds to 898,963 DIA-sets, a **template (tmpl)** image, **search (srch)** image, and their **difference (diff)**.*

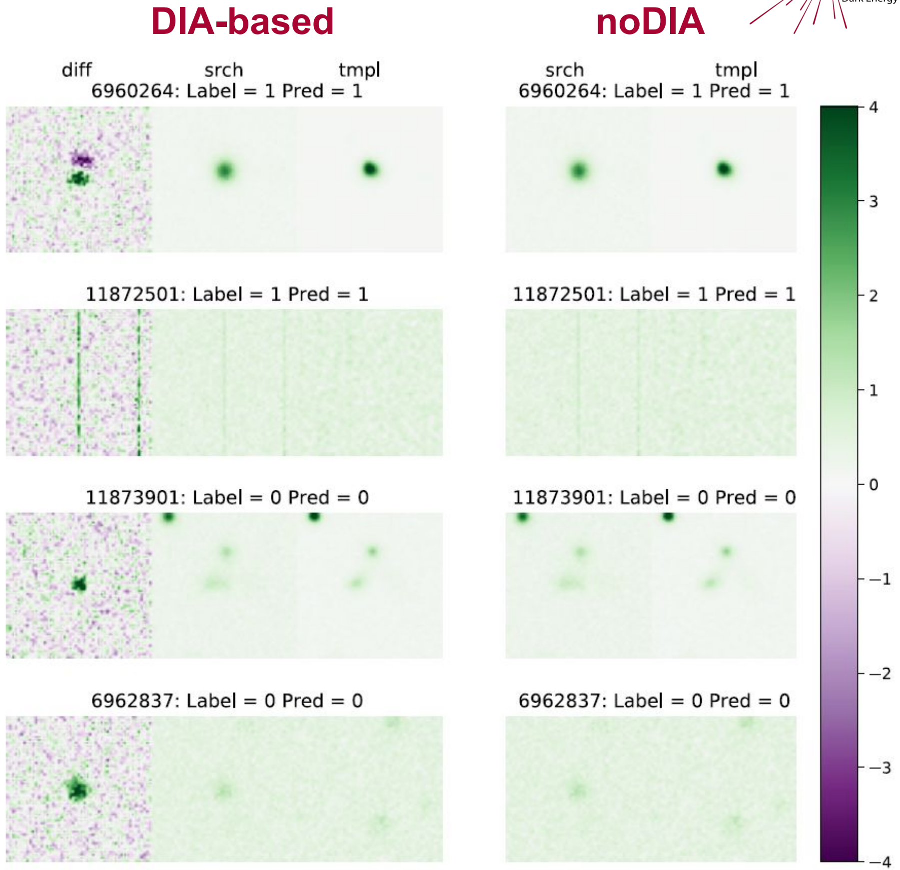
➔ Each image is 51x51 pixels, corresponding to approx 0.26 arcseconds square of sky.

*https://portal.nersc.gov/project/dessn/autoscan/
*https://www.darkenergysurvey.org/es/

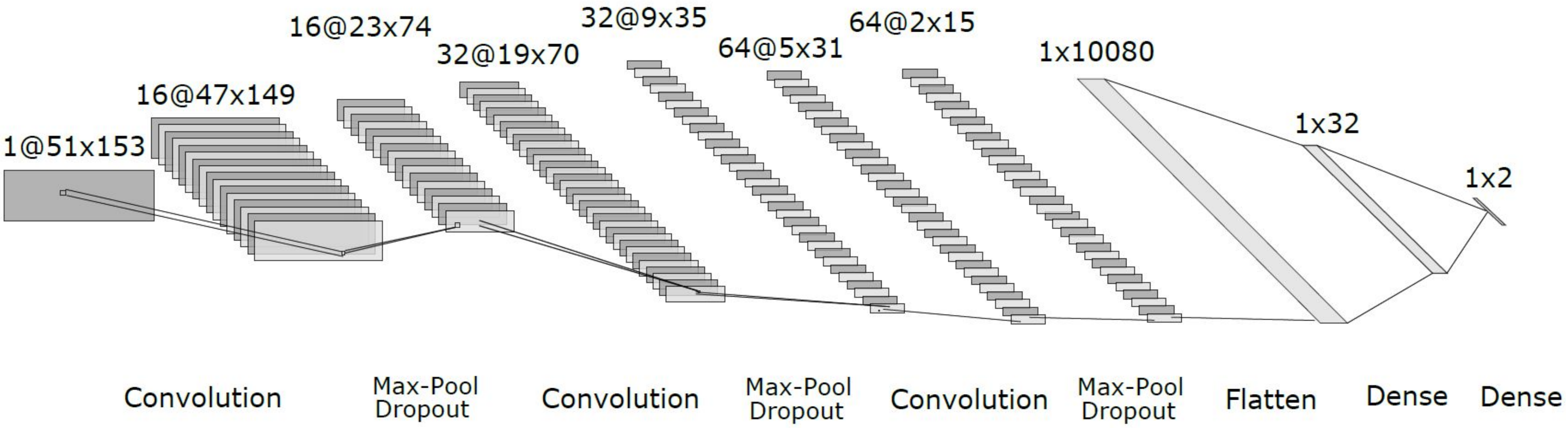
Two data sets to be train

➔ Mimic closely the way that human scanned this type of data for classification, and we will take advantage of this scheme when examining the models' decisions.

➔ Final shape of the data
DIA-based: (N, 51, 153)
noDIA: (N, 51, 102)



Architecture CNN: DIA-based



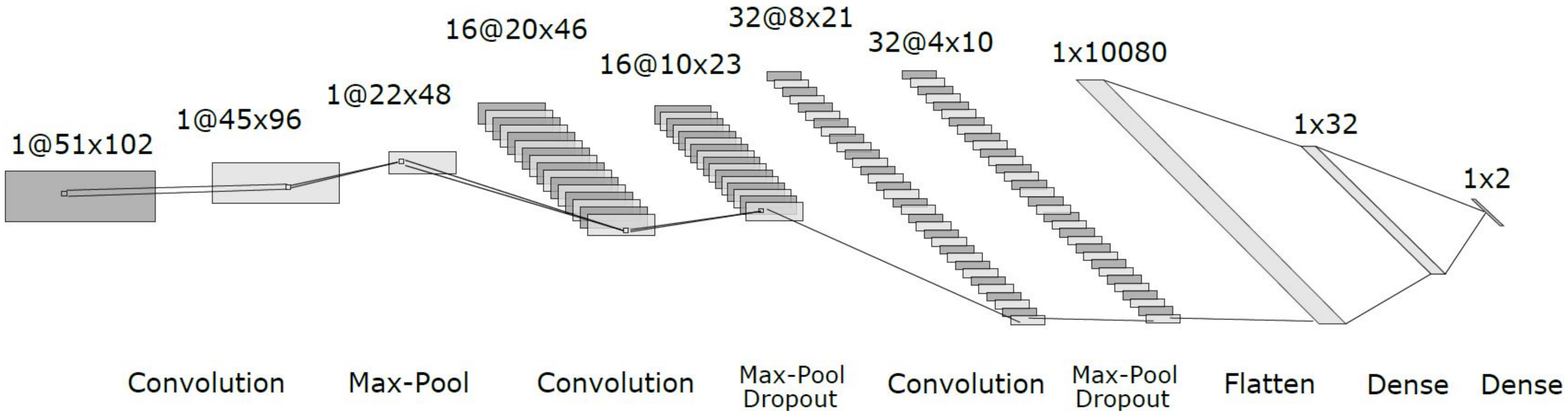
Architecture CNN: DIA-based

```
layer1 = keras.layers.Conv2D(16, kernel_size=(5, 5), padding="valid", activation="relu",
                             input_shape=(51,153,1))
layer2 = keras.layers.MaxPooling2D((2, 2), strides=2)
layer3 = keras.layers.Dropout(0.4)
layer4 = keras.layers.Conv2D(32, kernel_size=(5, 5), padding="valid", activation="relu")
layer5 = keras.layers.MaxPooling2D((2, 2), strides=2)
layer6 = keras.layers.Dropout(0.4)
layer7 = keras.layers.Conv2D(64, kernel_size=(5, 5), padding="valid", activation="relu")
layer8 = keras.layers.MaxPooling2D((2, 2), strides=2)
layer9 = keras.layers.Dropout(0.4)
layer10 = keras.layers.Flatten()
layer11 = keras.layers.Dense(32, activation="relu")
layer12 = keras.layers.Dense(2, activation="softmax")

opt = keras.optimizers.SGD(learning_rate=0.01)
model.compile(optimizer=opt, loss="sparse_categorical_crossentropy", metrics=["accuracy"])

history = model.fit(feat_tr2, targ_tr, validation_split=0.20, epochs=50, batch_size=20)
```

Architecture CNN: noDIA



Architecture CNN: noDIA

```
layer1 = keras.layers.Conv2D(1, kernel_size=(7, 7), padding="valid", activation="relu",
                             input_shape=(51,102,1))
layer2 = keras.layers.MaxPooling2D((2, 2), strides=2)
layer3 = keras.layers.Conv2D(16, kernel_size=(3, 3), padding="valid", activation="relu")
layer4 = keras.layers.MaxPooling2D((2, 2), strides=2)
layer5 = keras.layers.Dropout(0.4)
layer6 = keras.layers.Conv2D(32, kernel_size=(3, 3), padding="valid", activation="relu")
layer7 = keras.layers.MaxPooling2D((2, 2), strides=2)
layer8 = keras.layers.Dropout(0.4)
layer9 = keras.layers.Flatten()
layer10 = keras.layers.Dense(32, activation="relu")
layer11 = keras.layers.Dense(2, activation="softmax")
```

Train-Test split

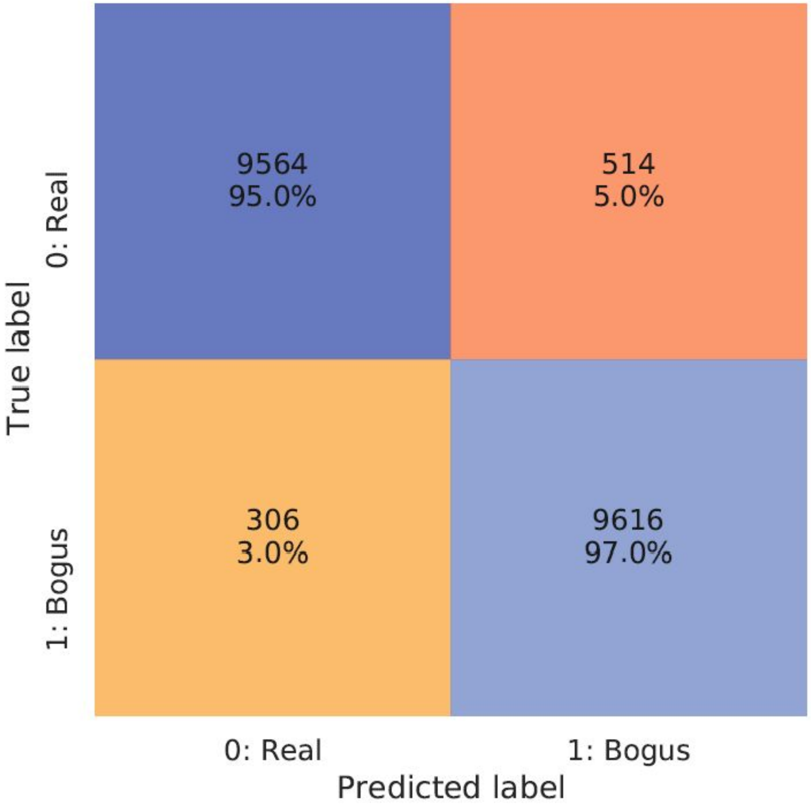
➔ The input of our Neural Networks are the horizontally stacked images of size 51x153 for the DIA-based model (diff, srch, tmpl), and 51x102 for noDIA model (srch, tmpl).

➔ For both the DIA-based model and the noDIA model 100□,000 images were used, **80,000** images for **training** and **20,000** for **validation.**

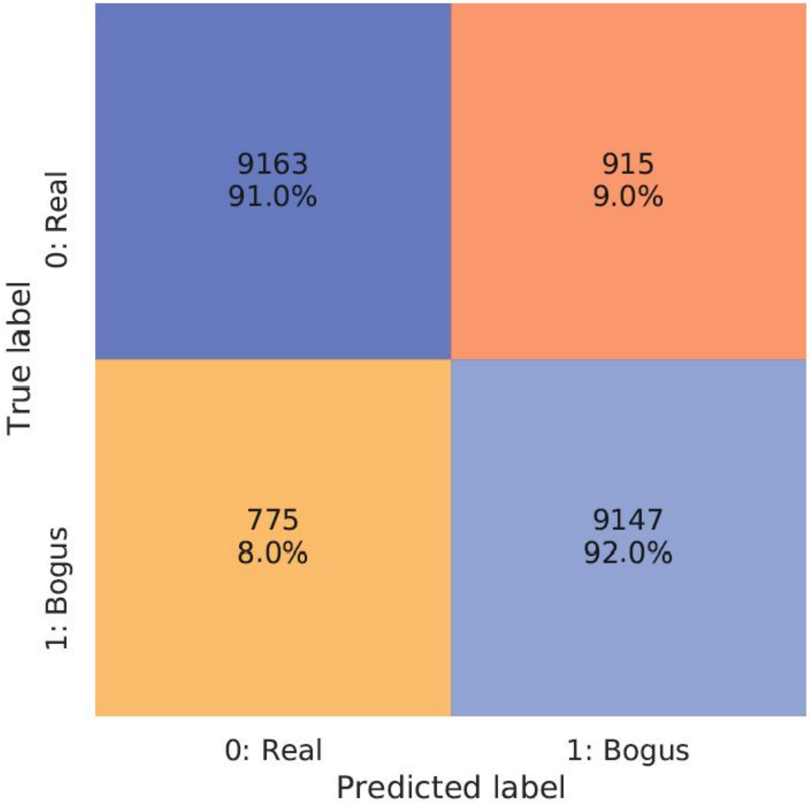
➔ The data is composed by **50,183** images labeled as **“bogus”** and **49,817** labeled as **“real”**.

Confusion matrices

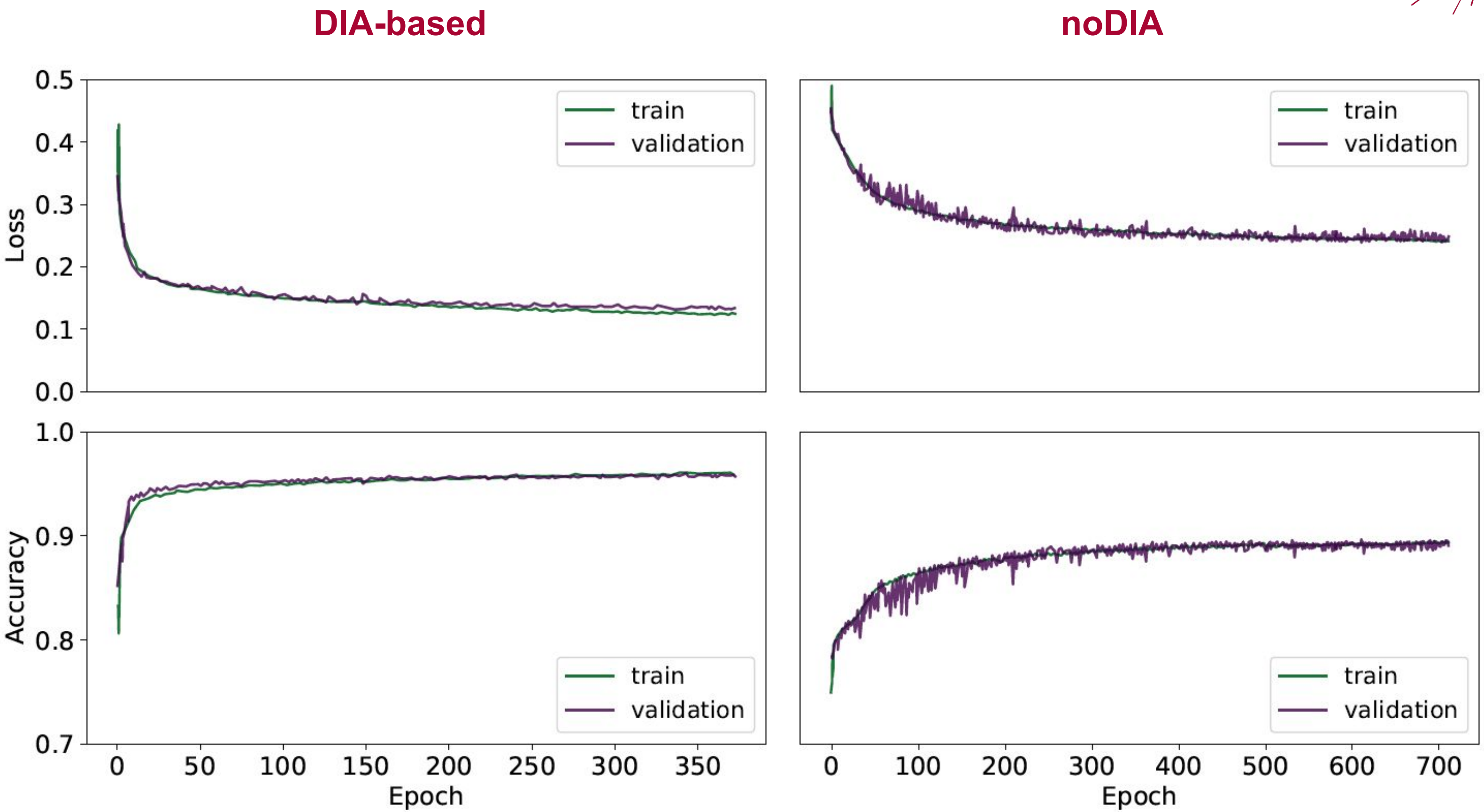
DIA-based



noDIA



Accuracy and loss plots



Summary until this point

- 96% accuracy for DIA-based model (so good!)

- Great potential for the noDIA with 92% accuracy (still so good!)
    - CNN architecture is not optimized for this input data.

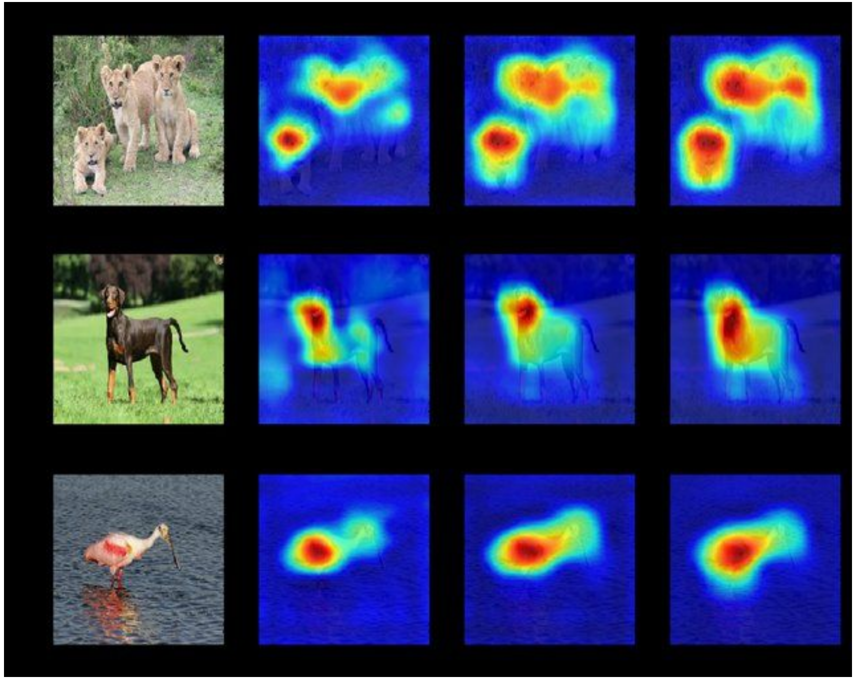
Now

➔ Interpretation of the results and CNNs outputs through the exploration of the Saliency maps.

Saliency maps

Saliency maps **quantify the importance of each pixel** of an image in input to a CNN in the training process.

They provide some level of interpretability through a process akin to feature importance analysis by enabling an assessment of which are **the pixels the model relies on the most for the final classification.**

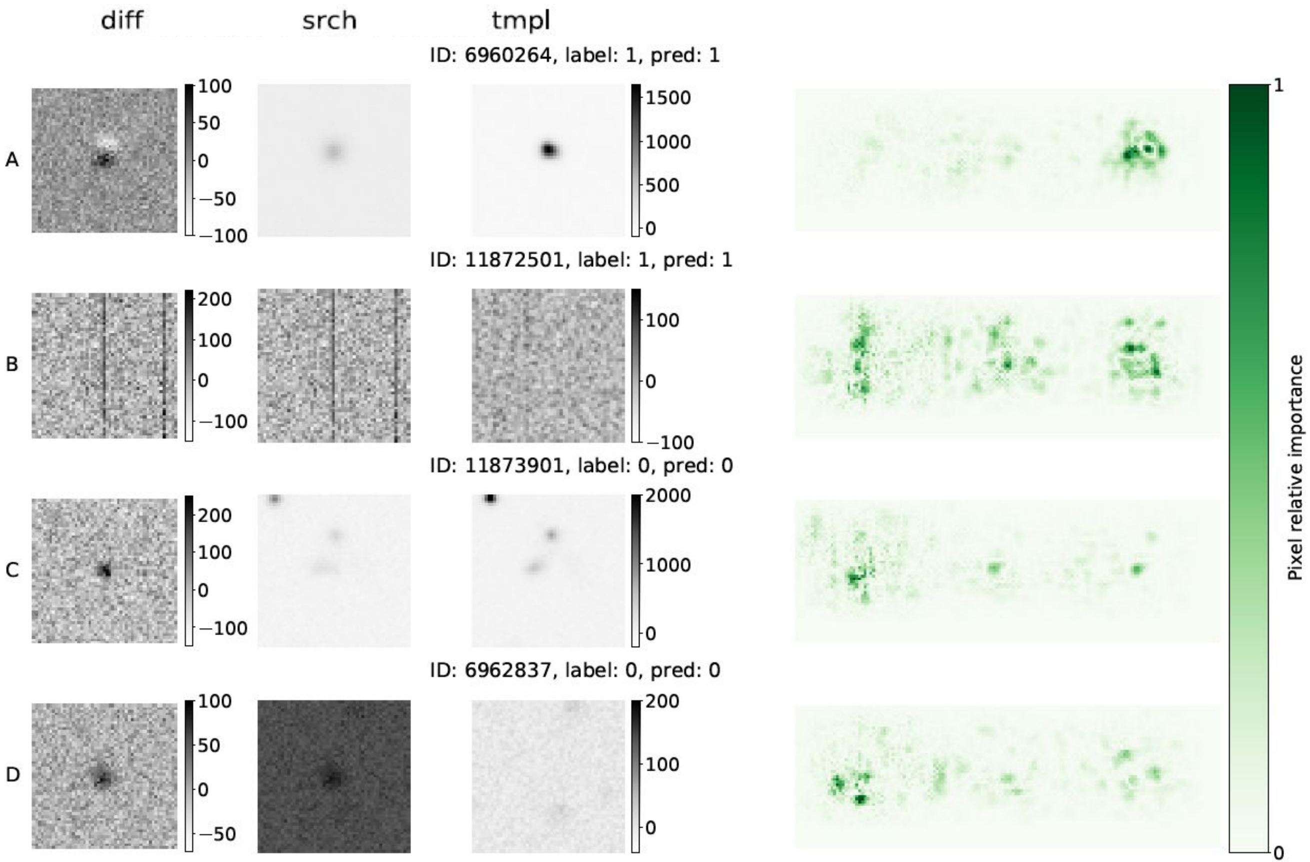


Saliency maps expectation

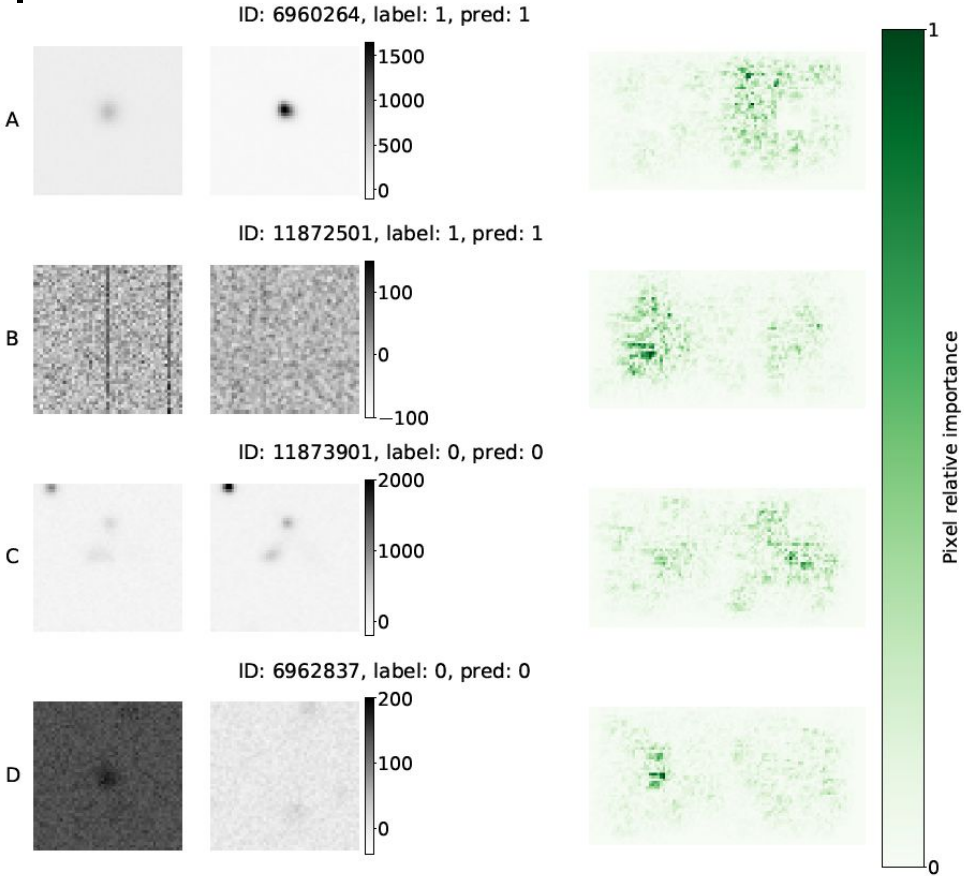
➔ DIA-based: a greater concentration of important pixels should be found in the diff image.

➔ noDIA: less clear, beyond our experience (intuition) as human scanner in the “real-bogus” classification.

Saliency maps: DIA-based



Saliency maps: noDIA



Conclusions

➔ **96% accuracy** for "real-bogus" classification using **DIA-based** CNN model.

➔ **92% accuracy** for "real-bogus" classification **without** using the **difference image** (the noDIA model)

➔ **CNN** trained with the **DIA output** primarily uses the information in the **diff** image to make the **final classification**,

➔ and that the model examines a diff-srch-tmpl image-set fundamentally differently in the cases where there is a transient, than in the cases where there is not one.

➔ The **noDIA model**, takes a more comprehensive **look at both tmpl and srch** images, but relies **primarily on tmpl** to enable the **reconstruction of the information found in the diff.**

Thank you!