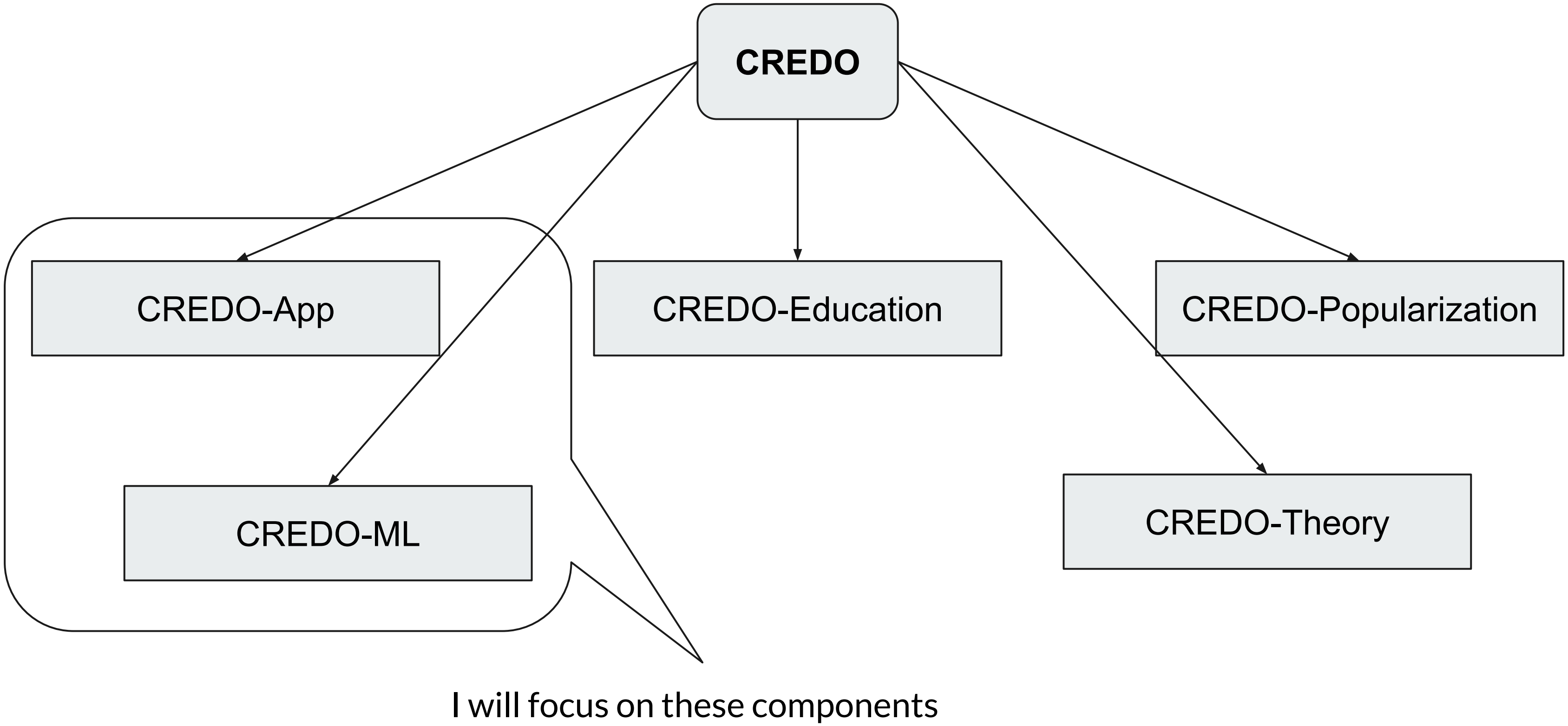
Status of the CREDO Detector and CREDO-ML

Łukasz Bibrzycki
Institute of Computer Science, Pedagogical University of Krakow
for CREDO-ML & CREDO App

Why this workshop ?

- boosting collaboration,
- setting immediate objectives,
- making competences inventory,
- discussing technicalities.

CREDO Structure

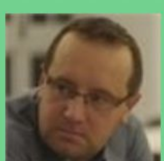


People and Institutions in CREDO-ML

Pedagogical University of Krakow

Olaf Bar

Łukasz Bibrzycki



Marcin Piekarczyk

Michał Frontczak

AGH University of Science and Technology

Krzysztof Rzecki

Tomasz Sośnicki

Cracow University of Techonolgy

Michał Niedźwiecki

Institute of Nuclear Physics PAS

Sławomir Stuglik

Piotr Homola

We meet every Friday at 12:00pm on Teams to discuss research progress. Everybody's invited !

People and Institutions in CREDO-App

Pedagogical University of Krakow

Bartek Pietras

Łukasz Bibrzycki

Marcin Piekarczyk

Michał Frontczak

Institute of Nuclear Physics PAS

Sławomir Stuglik

Piotr Homola

Cracow University of Techonolgy

Michał Niedźwiecki

Our co-authored papers

General papers

1. Cosmic Ray Extremely Distributed Observatory, CREDO Collaboration • Piotr Homola et al., Symmetry 12 (2020) 11, 1835

About: CREDO “Master paper”. Physics of interest, detectors, data management, public engagement and outreach.

2. Towards A Global Cosmic Ray Sensor Network: CREDO Detector as the First Open-Source Mobile Application Enabling Detection of Penetrating Radiation, CREDO Collaboration • Łukasz Bibrzycki, Dariusz Burakowski, Piotr Homola, Marcin Piekarczyk, Michał Niedźwiecki et al., Symmetry 12 (2020) 11, 1802

About: Complete description CREDO Detector. Overall design, detection algorithm, denoising, gamification, future developments.

Our co-authored papers

ML based analyses

1. CNN-Based Classifier as an Offline Trigger for the CREDO Experiment, Marcin Piekarczyk, Olaf Bar, Łukasz Bibrzycki, Michał Niedźwiecki, Krzysztof Rzecki et al., Sensors 21 (2021) 14, 4804

About: Artefact reduction by applying the Convolutional Neural Network to classify candidate hits.

2. Zernike Moment Based Classification of Cosmic Ray Candidate Hits from CMOS Sensors, Bar, O.; Bibrzycki, Ł.; Niedźwiecki, M.; Piekarczyk, M.; Rzecki, K.; Sośnicki, T.; Stuglik, S.; Frontczak, M.; Homola, P.; Alvarez-Castillo, D.E.; Andersen, T.; Tursunov, A., on behalf of CREDO Collaboration. Sensors **2021**, *21*, 7718.

About: Application of various statistical classifiers for distinguishing four main classes of Cosmic Ray signals: spots, tracks, worms and artefacts. Zernike moments were used as feature carriers.

Our co-authored papers

Spin off collaborations

1. Recognition of Cosmic Ray Images Obtained from CMOS Sensors Used in Mobile Phones by Approximation of Uncertain Class Assignment with Deep Convolutional Neural Network, Hachaj T, Bibrzycki Ł, Piekarczyk M. *Sensors.* 2021; 21(6):1963

About: thresholding, approximation of annotators' assignment

2. Deep neural network architecture for low-dimensional embedding and classification of cosmic ray images obtained from CMOS cameras Hachaj T., Piekarczyk M., Bibrzycki Ł., paper accepted for The 28th International Conference on Neural Information Processing (ICONIP 2021)

About: Low dimensional hit image embedding, complete classification of entire CREDO data set.

Our co-authored papers

Strong participation in ICRC 2021

1. Machine learning aided noise filtration and signal classification for CREDO experiment CREDO Collaboration • Łukasz Bibrzycki et al., PoS ICRC2021 (2021), 227

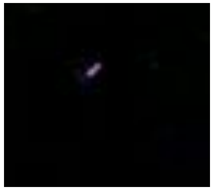
About: application of wavelets, review of methods applied so far.

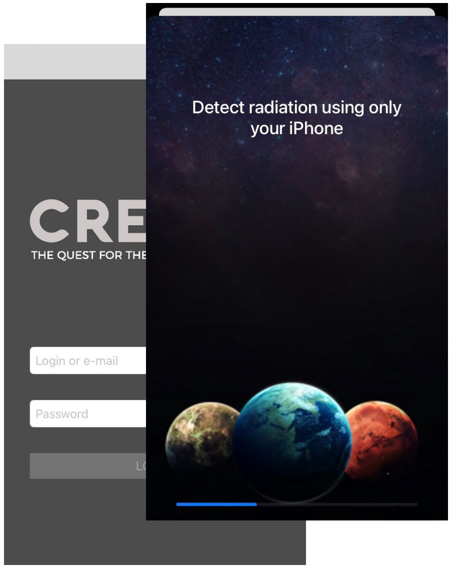
Co-authoring several other papers submitted at ICRC 2021.

CREDO Detector on the iOS and Android platforms

CREDO App status - iOS

- CREDO Detector for iOS is coming soon!
- v1.0 - Waiting for approval and release in App Store!
- App based on Cosmic Ray - Detect Radiation by Tom Andersen.
- Michał Frontczak is here the person in charge.

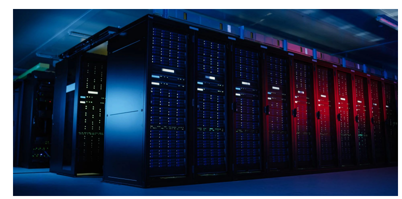


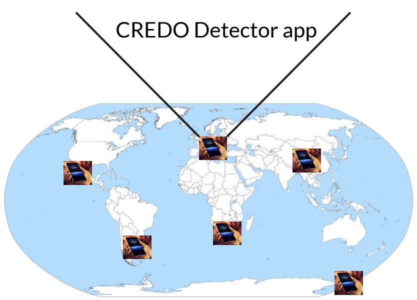


CREDO App status - Android v 1.2

CREDO Detector for Android - v1.2:

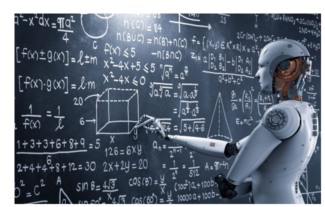
- operational and collecting data





64x64px images,
timestamp,
GPS location,
detector orientation by
accelerometer and compass

storage data in database



Smartphones around the world

data analysis by CREDO-ML team

CREDO App status - Android v 2.0

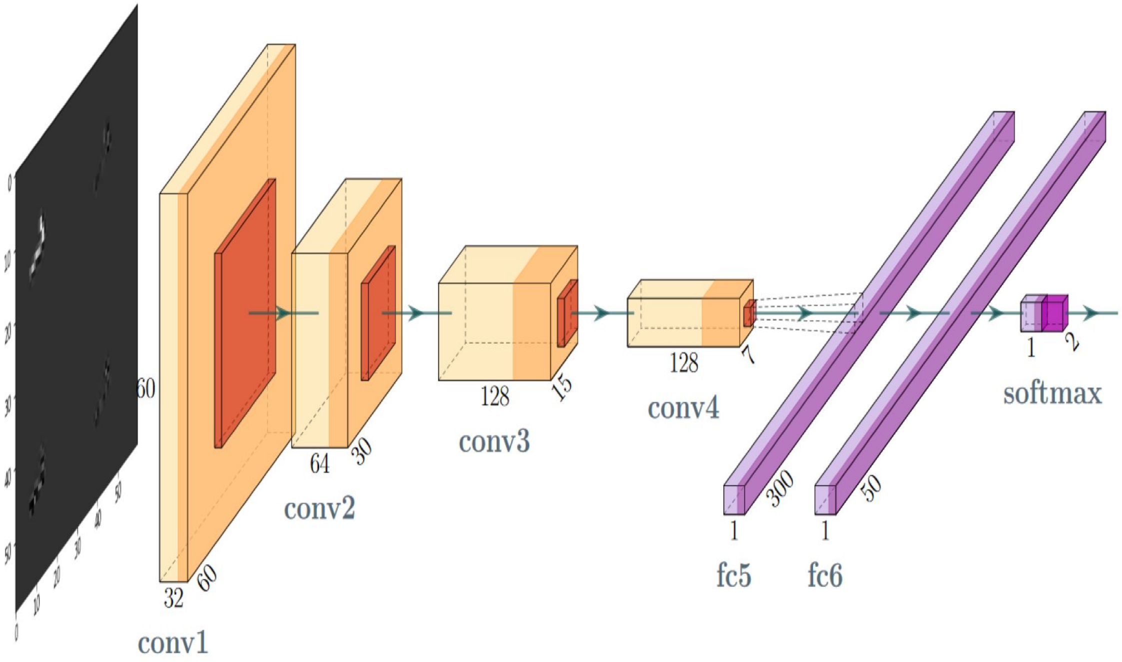
- Limitations of the standard Android picture taking procedure
- Entirely new version is developed from scratch which is based on the low level RAW format
- Bartek Pietras is a person in charge

Machine Learning based candidate hit analyses

CREDO-ML status - CNN based trigger

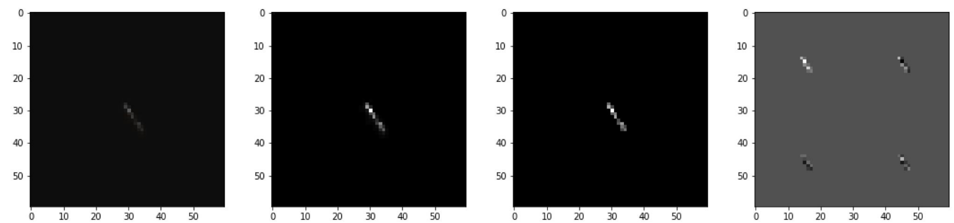
- the need for artefact rejection
- general idea: we look only at morphology
- architecture
- application of wavelets (to amplify signal properties)
- results: trigger efficiency, examples of accepted and rejected hits
- open problems: including temporal properties of the signal (too frequent hits)

CNN-based trigger :: concept

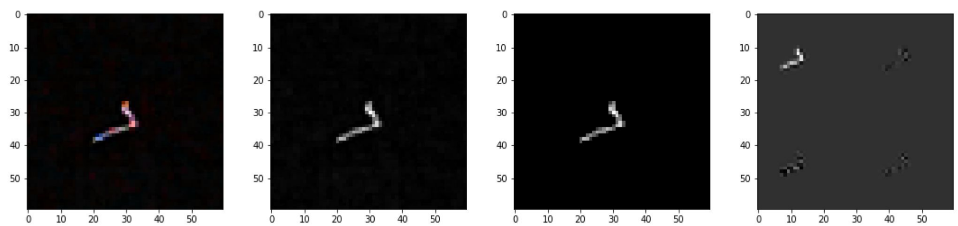


- Morphology/shape oriented recognition/classification
- Deep convolutional network tuned to the nature and size of the data
- Three variants of input preprocessing:
  - raw data,
  - wavelet transformed data,
  - combined approach (raw + wavelet)

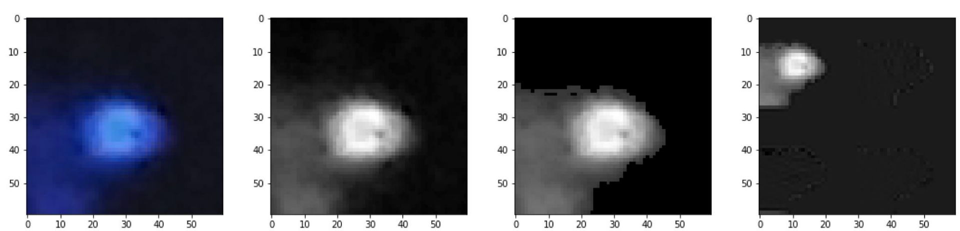
CNN-based trigger :: preprocessing



(**a**) Color (**b**) Grayscale (**c**) Threshold (**d**) Wavelet



(**a**) Color (**b**) Grayscale (**c**) Threshold (**d**) Wavelet



(**a**) Color (**b**) Grayscale (**c**) Threshold (**d**) Wavelet

General preprocessing flow:

(**a**) color input image,

(**b**) grayscale accumulation,

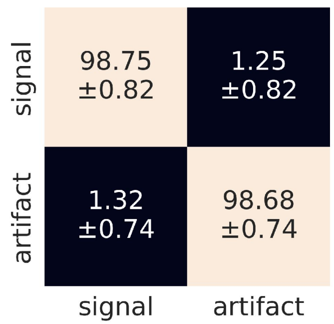
(**c**) adaptive thresholding,

(**d**) wavelet transformation.

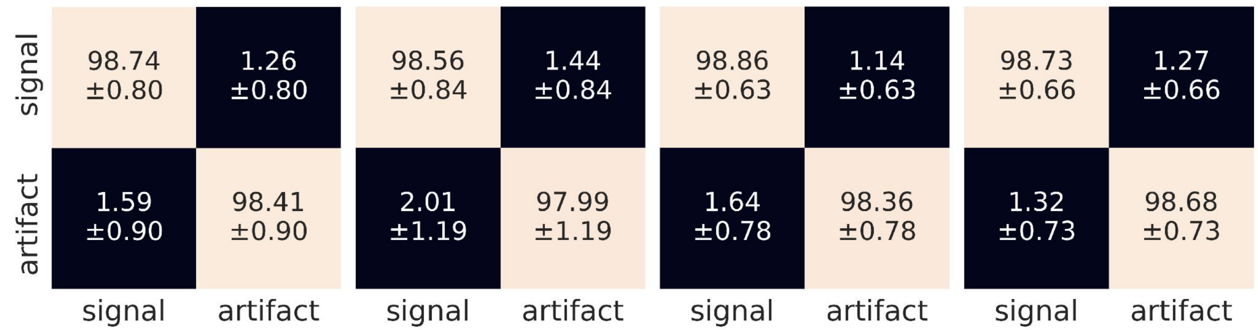
Examples for: track-type image, worm-type image and artefact image.

CNN-based trigger :: performance

Raw data

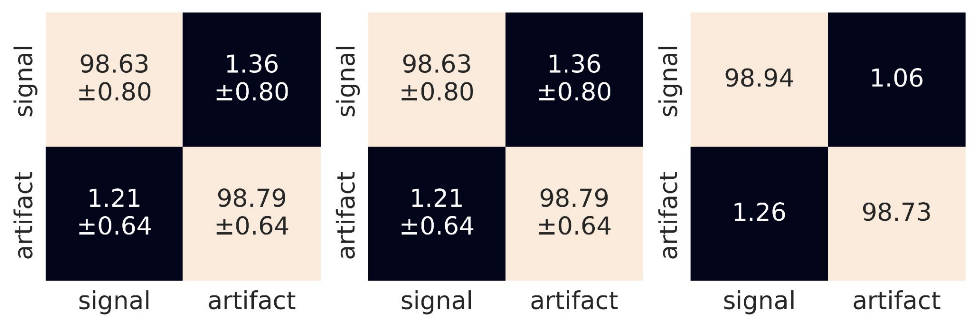


Wavelet transformed data



(**a**) D2 (**b**) D20 (**c**) D2:D4 (**d**) D2:D20

Combined approach



(**a**) raw+D2 (**b**) raw+D20 (**c**) raw+D2:D20

CREDO-ML status - Multi class hit classification

- we focus on the morphology exclusively to distinguish spots, tracks, worms and artefacts,
- preprocessing,
- Zernike moments as features provide the link to commonly used astrophysical signal properties like ellipticity or solidicity,

ML classifiers used in experiments

- DTC – decision tree classifier
- GNB – Gaussian Naive Bayes (GaussianNB) based classifier
- KNN – k-nearest neighbors classifier
- LDA – Linear Discriminant Analysis based classifier
- LRC – Logistic Regression (aka logit, MaxEnt) based classifier
- LSV – Linear Support Vector Classification - SVM based classifier
- MLP – Multi-layer Perceptron based classifier
- NSV – nu-Support Vector Classification based classifier
- QDA – Quadratic Discriminant Analysis based classifier
- SGD – Linear SVM based classifier with SGD training
- SVC – C-Support Vector Classifier with radial basis function - RBF as a kernel

Accuracy achieved for ML classifiers

	1st Phase		2nd Phase	
Classifier	CV	Test	Mean30	Std30
DTC	0.8037	0.7966	0.8194	0.0183
GNB	0.4764	0.7191	0.7017	0.0216
KNN	0.8415	0.7883	0.7808	0.0188
LDA	0.6534	0.5723	0.5538	0.0240
LRC	0.8847	0.8470	0.8292	0.0178
LSV	0.8449	0.8050	0.7843	0.0222
MLP	**0.9154**	**0.8784**	**0.8799**	0.0134
NSV	**0.9016**	**0.9015**	**0.8711**	0.0136
QDA	0.6999	0.7631	0.7422	0.0209
SGD	0.8683	0.8008	0.8027	0.0202
SVC	**0.9177**	**0.8952**	**0.8818**	0.0124

Ensembled classifiers used in experiments

- ETC – An extra-trees classifier
- GBC – Gradient Boosting for classification
- RFC – random forest classifier
- BAG – Bagging classifier
- OvO – classifier that implements one-vs.-one multiclass strategy
- OvR – classifier that implements the one-vs.-rest multiclass strategy
- VOT – hard voting based classifier

Accuracy achieved for Ensembled classifiers

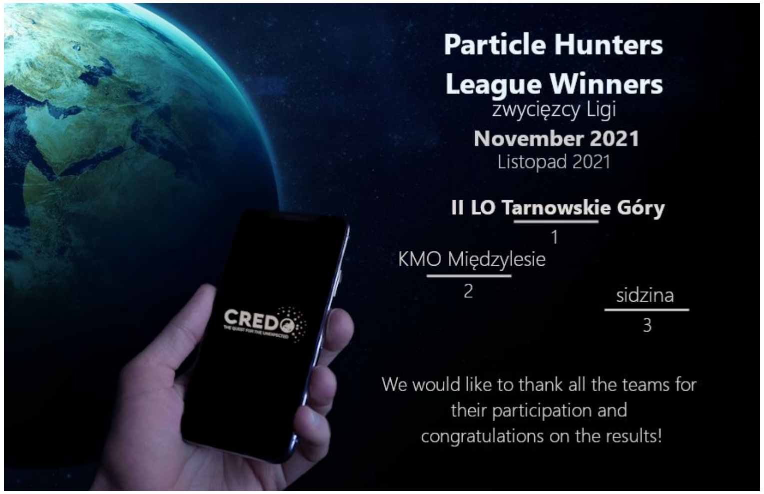
	1st Phase		2nd Phase	
Classifier	**CV**	**Test**	**Mean30**	**Std30**
ETC	0.8986	0.8973	0.8758	0.0142
GBC	0.8950	0.8847	0.8741	0.0150
RFC	0.8853	0.8763	0.8699	0.0137
VOT	0.9205	0.8973	0.8841	0.0123
BAG/SVC	0.9078	0.8868	0.8805	0.0135
OvO/MLP	0.9101	0.8973	0.8880	0.0145
OvO/SVC	0.9171	0.8889	0.8850	0.0148
OvR/MLP	0.9138	0.8952	0.8853	0.0139

Applications

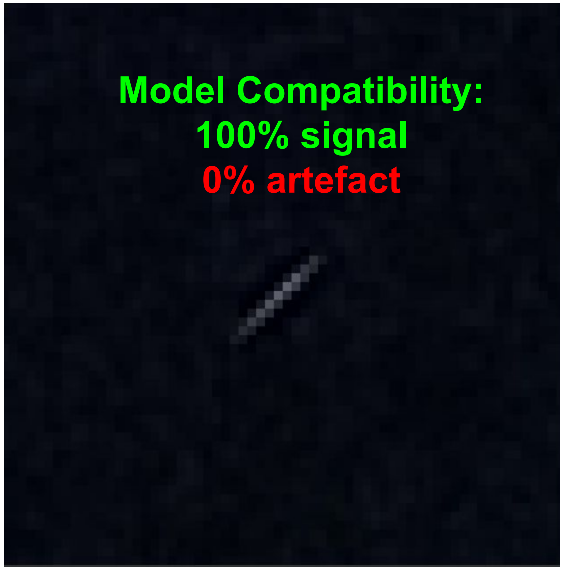
“Particle Hunters”

The 4th edition of the "Particle Hunters" contest is currently running.

- The goal of the contest - increasing the CREDO Detector user base.
- Application of Machine Learning to distinguish signals from artefacts, and to pre-classify particles (dots, tracks, worms).
- This method replaces the filters that were used in previous editions.
- Triggers work in the ensemble mode so the final assignment (signal vs. artefact) is taken by majority vote

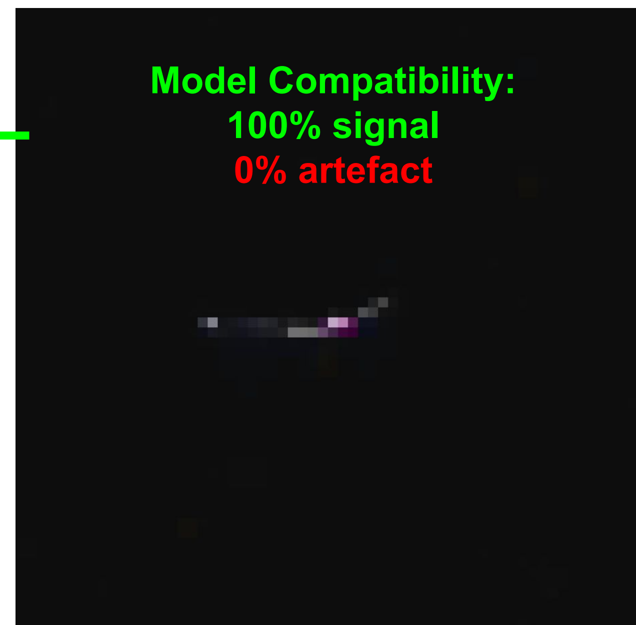


Multi class hit classifications - signals examples

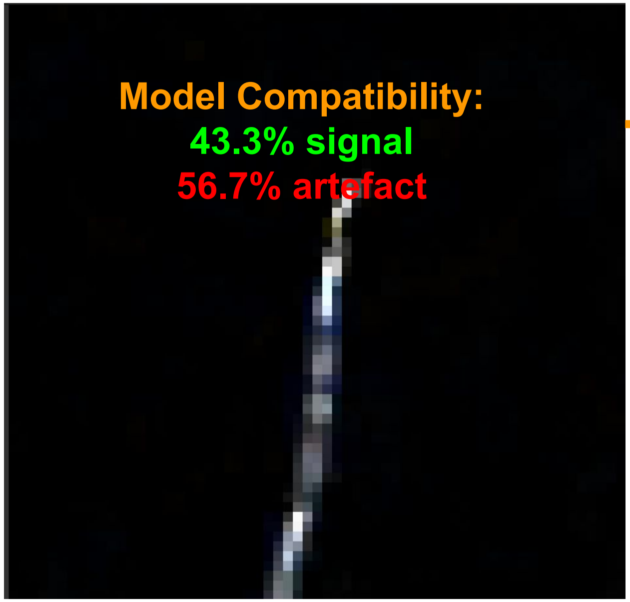


anti_artefact	0,signal	CNN_big_w20	0,signal	nsv	1,signal
CNN_small_w0	0,signal	CNN_small_w2_20	0,signal	ovo_mlp	1,signal
CNN_big_w0	0,signal	CNN_big_w2_20	0,signal	ovo_svm	1,signal
CNN_small_w2	0,signal	CNN_small_raw	0,signal	ovr_mlp	1,signal
CNN_big_w2	0,signal	CNN_big_raw	0,signal	rf	1,signal
CNN_small_w0_2	0,signal	etc	1,signal	svm	1,signal
CNN_big_w0_2	0,signal	bagsvc	1,signal	vot	1,signal
CNN_small_w2_10	0,signal	gbc	1,signal	baseline	0,signal
CNN_big_w2_10	0,signal	knn	1,signal	baseline_knn	1,signal
CNN_small_w20	0,signal	mlp	1,signal	baseline_rf	1,signal

anti_artefact	0,signal	CNN_big_w20	0,signal	nsv	1,signal
CNN_small_w0	0,signal	CNN_small_w2_20	0,signal	ovo_mlp	1,signal
CNN_big_w0	0,signal	CNN_big_w2_20	0,signal	ovo_svm	1,signal
CNN_small_w2	0,signal	CNN_small_raw	0,signal	ovr_mlp	2,signal
CNN_big_w2	0,signal	CNN_big_raw	0,signal	rf	2,signal
CNN_small_w0_2	0,signal	etc	1,signal	svm	1,signal
CNN_big_w0_2	0,signal	bagsvc	1,signal	vot	2,signal
CNN_small_w2_10	0,signal	gbc	1,signal	baseline	0,signal
CNN_big_w2_10	0,signal	knn	1,signal	baseline_knn	1,signal
CNN_small_w20	0,signal	mlp	2,signal	baseline_rf	1,signal

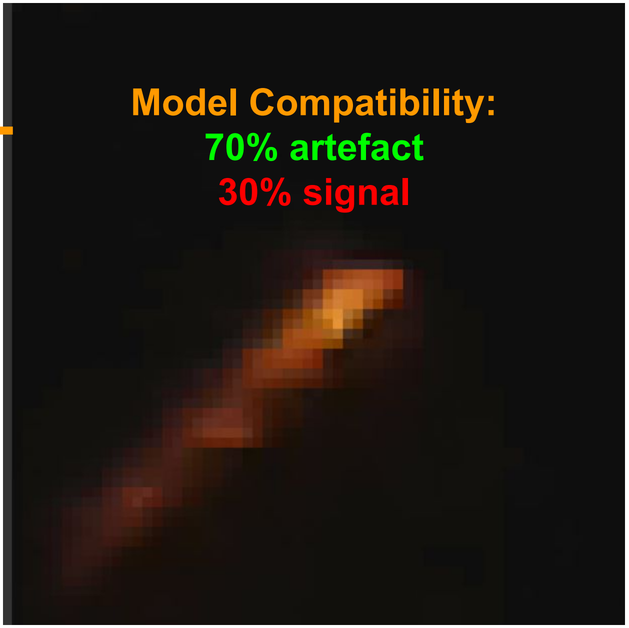


Multi class hit classifications - problematic images

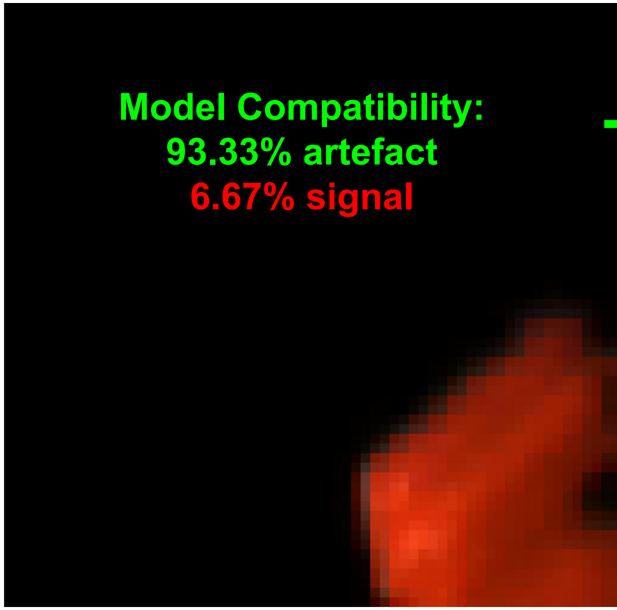


anti_artefact	1,artefact	**CNN_big_w20**	1,artefact	**nsv**	3,artefact
CNN_small_w0	1,artefact	**CNN_small_w2_20**	1,artefact	**ovo_mlp**	2,signal
CNN_big_w0	1,artefact	**CNN_big_w2_20**	1,artefact	**ovo_svm**	2,signal
CNN_small_w2	1,artefact	**CNN_small_raw**	1,artefact	**ovr_mlp**	2,signal
CNN_big_w2	1,artefact	**CNN_big_raw**	1,artefact	**rf**	2,signal
CNN_small_w0_2	0,signal	**etc**	2,signal	**svm**	2,signal
CNN_big_w0_2	1,artefact	**bagsvc**	2,signal	**vot**	2,signal
CNN_small_w2_10	0,signal	**gbc**	2,signal	**baseline**	1,signal
CNN_big_w2_10	1,artefact	**knn**	3,artefact	**baseline_knn**	3,artefact
CNN_small_w20	1,artefact	**mlp**	2,signal	**baseline_rf**	3,artefact

anti_artefact	0,signal	**CNN_big_w20**	1,artefact	**nsv**	2,signal
CNN_small_w0	1,artefact	**CNN_small_w2_20**	1,artefact	**ovo_mlp**	3,artefact
CNN_big_w0	1,artefact	**CNN_big_w2_20**	1,artefact	**ovo_svm**	2,signal
CNN_small_w2	1,artefact	**CNN_small_raw**	1,artefact	**ovr_mlp**	2,signal
CNN_big_w2	1,artefact	**CNN_big_raw**	1,artefact	**rf**	3,artefact
CNN_small_w0_2	1,artefact	**etc**	3,artefact	**svm**	2,signal
CNN_big_w0_2	1,artefact	**bagsvc**	3,artefact	**vot**	2,signal
CNN_small_w2_10	1,artefact	**gbc**	3,artefact	**baseline**	1,signal
CNN_big_w2_10	1,artefact	**knn**	2,signal	**baseline_knn**	3,artefact
CNN_small_w20	1,artefact	**mlp**	2,signal	**baseline_rf**	3,artefact

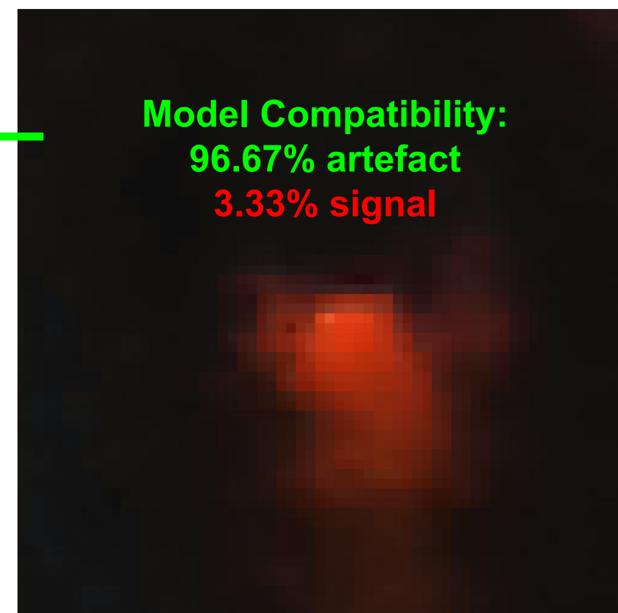


Multi class hit classifications - artefacts examples



anti_artefact	1,artefact	**CNN_big_w20**	1,artefact	**nsv**	3,artefact
CNN_small_w0	1,artefact	**CNN_small_w2_20**	1,artefact	**ovo_mlp**	3,artefact
CNN_big_w0	1,artefact	**CNN_big_w2_20**	1,artefact	**ovo_svm**	3,artefact
CNN_small_w2	1,artefact	**CNN_small_raw**	1,artefact	**ovr_mlp**	3,artefact
CNN_big_w2	1,artefact	**CNN_big_raw**	0,signal	**rf**	3,artefact
CNN_small_w0_2	1,artefact	**etc**	3,artefact	**svm**	3,artefact
CNN_big_w0_2	1,artefact	**bagsvc**	3,artefact	**vot**	3,artefact
CNN_small_w2_10	1,artefact	**gbc**	3,artefact	**baseline**	1,signal
CNN_big_w2_10	1,artefact	**knn**	3,artefact	**baseline_knn**	3,artefact
CNN_small_w20	1,artefact	**mlp**	3,artefact	**baseline_rf**	3,artefact

anti_artefact	1,artefact	**CNN_big_w20**	1,artefact	**nsv**	3,artefact
CNN_small_w0	1,artefact	**CNN_small_w2_20**	1,artefact	**ovo_mlp**	3,artefact
CNN_big_w0	1,artefact	**CNN_big_w2_20**	1,artefact	**ovo_svm**	3,artefact
CNN_small_w2	1,artefact	**CNN_small_raw**	1,artefact	**ovr_mlp**	3,artefact
CNN_big_w2	1,artefact	**CNN_big_raw**	1,artefact	**rf**	3,artefact
CNN_small_w0_2	1,artefact	**etc**	3,artefact	**svm**	3,artefact
CNN_big_w0_2	1,artefact	**bagsvc**	3,artefact	**vot**	3,artefact
CNN_small_w2_10	1,artefact	**gbc**	3,artefact	**baseline**	1,signal
CNN_big_w2_10	1,artefact	**knn**	3,artefact	**baseline_knn**	3,artefact
CNN_small_w20	1,artefact	**mlp**	3,artefact	**baseline_rf**	3,artefact



CREDO data dissemination

Web application is under development to show:

- hit image,
- image id,
- class assignments by individual classifiers,
- class eventually assigned after voting,
- user id,
- geographical coordinates of the hit,
- timestamp,
- number of 60x60 images with a given timestamp on device,
- others,...

Outlook

- need for 10x larger training dataset (how to annotate the data efficiently and consistently),
- actively seek for new users social media, schools, more international - we need **much** more data,
- open problems: include metadata in training, like timestamp, geolocation, detector orientation.
- symulacje pęków, sprawdzić jakie nasycenie detektorów jest potrzebne, wstępne badania by CRAYFIS Extensive
- Air Shower simulation:
    - required detector density estimation (preliminary research by CRAYFIS),
    - required statistics,
    - sociology of the experiment (sustaining the users' interest),

Thank you for your attention !

Back-up slide - Sławek

- W przypadku klarownych przypadków widać dużą zgodność wszystkich modeli ponad 90%.
- Przypadki detekcji będących na pograniczu sygnał/artefact mogą jeszcze stanowić problem dla poprawnego rozpoznawania. Da się dostrzec też zależność “modelową” w której to przy konkretnym przypadku zawodzi cały model, np CNN dla 1 przykładu trudnych przypadków czy STD dla drugiego przypadku.
- Ciekawe, że w większości (⅚) przypadków stary filtr antyartefaktowy określał poprawnie przypadek.
- porównanie tych trzech grup przypadków wskazuje na to, że ten temat jeszcze nie jest zakończony, że modele muszą być douczone i koniecznie dodane muszą być w dużej liczbie do setu treningowego przypadki “pograniczne”
- Nie możemy ignorować kolorów śladów. Jednolity odcień koloru (pomarańczowy, czerwony, niebieski, zielony itp) w dużej liczbie pikseli wskazuje na artefact.

To są moje uwagi/przemyślenia - można o nich wspomnieć jeśli się z którymś zgadzasz. Ale finalnie usunąć tą strone komentarzową.

CREDO App status Android v 1.2

v.2.0 under development

stuff from Bartek Pietras

histogram pokazujący akumulację szumu w funkcji narastającego czasu ekspozycji