

Modelowanie niepewności w procesie klasyfikacji przy użyciu głębokiej sieci neuronowej: zastosowanie do rozpoznawania cząstek promieniowania kosmicznego na zdjęciach z kamer CMOS

Tomasz Hachaj

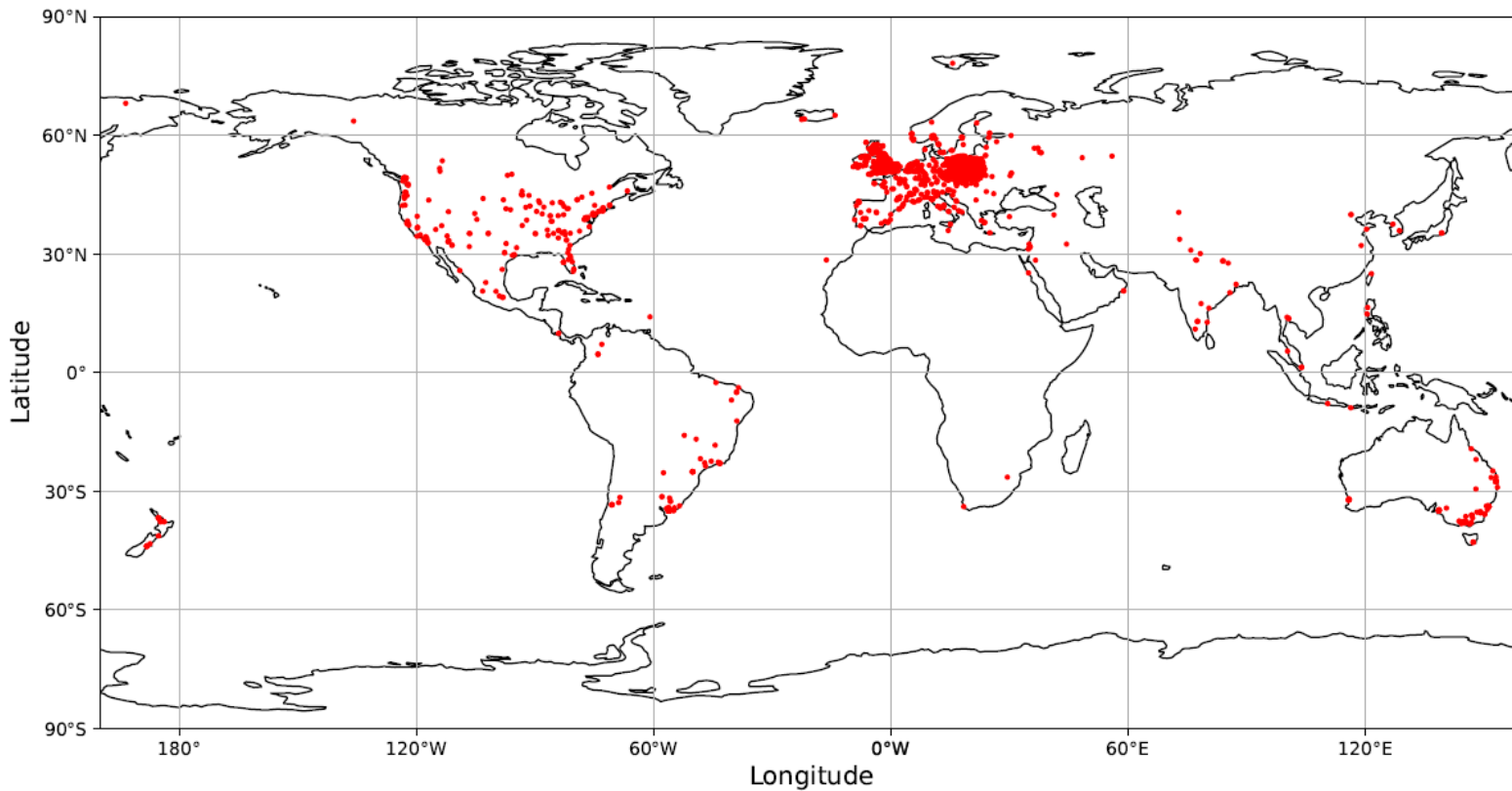
Instytut Informatyki Uniwersytetu Pedagogicznego w Krakowie

tomasz.hachaj@up.krakow.pl

<https://sppr.up.krakow.pl/hachaj/>



Projekt CREDO



P. Homola et. al., Cosmic Ray Extremely Distributed Observatory, October 2020, DOI: [10.20944/preprints202010.0298.v1](https://doi.org/10.20944/preprints202010.0298.v1)



Problem

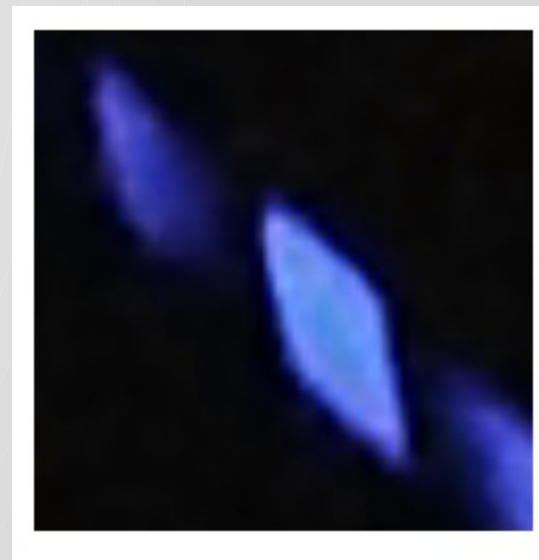
- Dany jest zbiór zdjęć cząstek promieniowanie kosmicznego, które wykrywane są przy pomocy kamer CMOS. Na podstawie parametrów morfologicznych obrazów uzyskanych po uderzeniu cząstki w detektor CMOS, chcemy przypisać uzyskane zobrazowanie do jednej z czterech klas:
- Kropka (spot)
- Linia (track)
- Robak (worm)
- Artefakt (artefact)
- Zbiór został wstępnie opisany (każdemu zdjęciu została nadana etykieta) przez grupę osób oceniających każde zobrazowanie (sędziów).
- Każdy sędzia przypisywał zdjęcie do jednej z powyższych klas, mógł również stwierdzić, że dany przypadek jest niejednoznaczny i nie dokonać przypisania do żadnej z tych klas.
- Używając dotychczasowo opisanych danych należy stworzyć algorytm, pozwalający na ocenę nieanalizowanych wcześniej zdjęć.
- Jest to problem klasyfikacyjny w oparciu o niepewny zbiór treningowy.



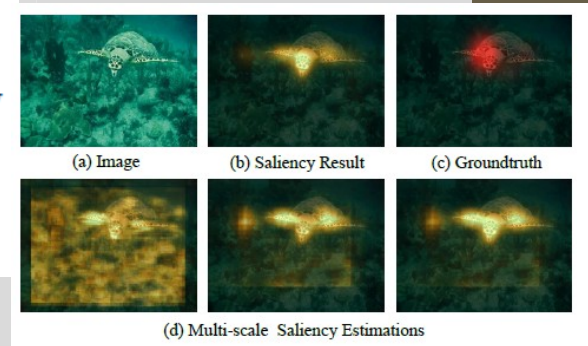
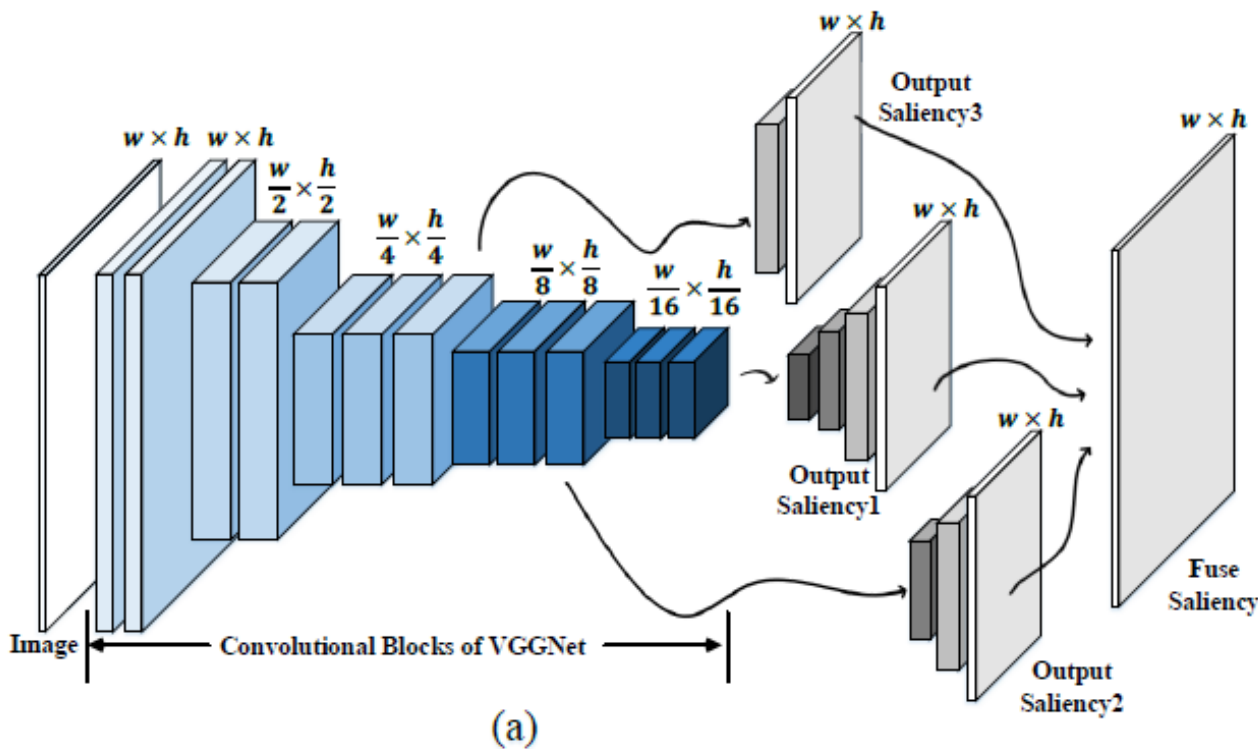
Zbiór danych

- Zbiór dostępny jest pod adresem: <https://credo.nkg-mn.com/>
- Pracowałem na zbiorze ocenionym przez 5 sędziów, zbiór został dodatkowo uzupełniony o pewną liczbę artefaktów, Ostatecznie zawierał 2350 obrazów.
- Oceny sędziów miały następującą dystrybucję:

	0	1	2	3	4	5
Spots	1790	103	82	34	80	261
Tracks	1832	136	91	73	109	109
Worms	1834	198	85	98	104	31
Artefacts	1115	82	3	0	0	1150



Zastosowanie głębokich sieci neuronowych do ekstrakcji cech



- Wenguan Wang, Jianbing Shen, Deep Visual Attention Prediction, IEEE Trans Image Process, 2018 May;27(5):2368-2378, doi: 10.1109/TIP.2017.2787612. Epub 2017 Dec 27

Sieci pretrenowane – transfer learning

IM GENET








SEARCH

[Home](#) | [Explore](#)
[About](#) | [Download](#)

14,197,122 images, 21841 synsets indexed

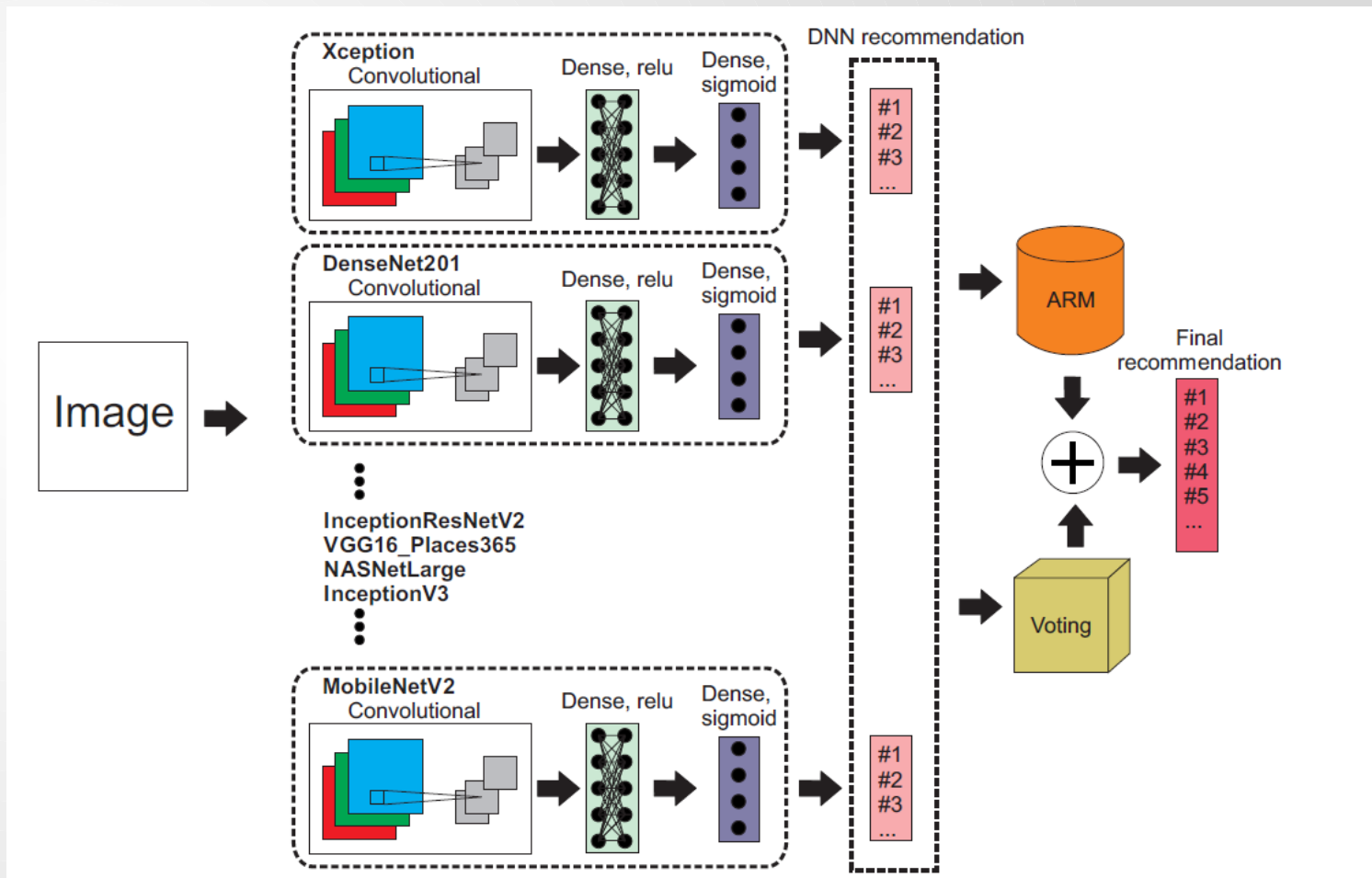
Not logged in. [Login](#) | [Signup](#)

ImageNet is under maintenance. Only ILSVRC synsets are included in the search results. Rank by [Popularity](#) [Depth](#)





	<p>Synset: tiger cat Definition: a cat having a striped coat. <i>Popularity percentile:</i> 78% <i>Depth in WordNet:</i> 8</p>
	<p>Synset: lesser panda, red panda, panda, bear cat, cat bear, Ailurus fulgens Definition: reddish-brown Old World raccoon-like carnivore; in some classifications considered unrelated to the giant pandas. <i>Popularity percentile:</i> 68% <i>Depth in WordNet:</i> 12</p>
	<p>Synset: Egyptian cat Definition: a domestic cat of Egypt. <i>Popularity percentile:</i> 67% <i>Depth in WordNet:</i> 8</p>
	<p>Synset: Persian cat Definition: a long-haired breed of cat. <i>Popularity percentile:</i> 59% <i>Depth in WordNet:</i> 8</p>
	<p>Synset: tabby, tabby cat Definition: a cat with a grey or tawny coat mottled with black. <i>Popularity percentile:</i> 58% <i>Depth in WordNet:</i> 8</p>
	<p>Synset: Siamese cat, Siamese Definition: a slender short-haired blue-eyed breed of cat having a pale coat with dark ears paws face and tail tip. <i>Popularity percentile:</i> 57% <i>Depth in WordNet:</i> 8</p>
	<p>Synset: Madagascar cat, ring-tailed lemur, Lemur catta Definition: small lemur having its tail barred with black. <i>Popularity percentile:</i> 45% <i>Depth in WordNet:</i> 12</p>

- <http://www.image-net.org/>

Problem klasyfikacji obiektu do kilku potencjalnych klas



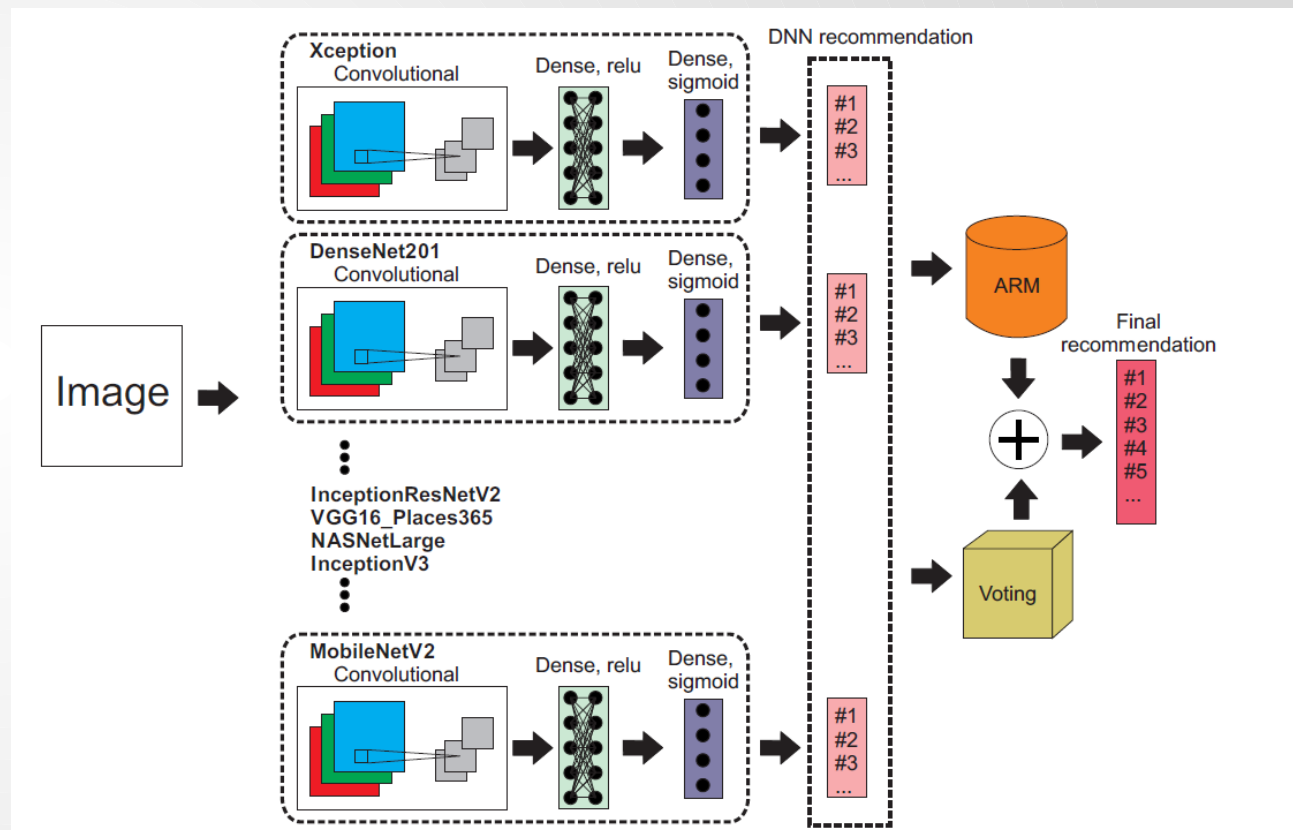
Problem klasyfikacji obiektu do kilku potencjalnych klas

			
illustration (318)	art (45)	architecture (181)	ootd (94)
drawing (137)	coffee (135)	hollywood (698)	son (502)
boyfriend (30)	tired (13)	vintage (130)	boy (53)
		design (98)	instagood (17)
		losangeles (397)	love (0)
		california (259)	
		yellow (4)	

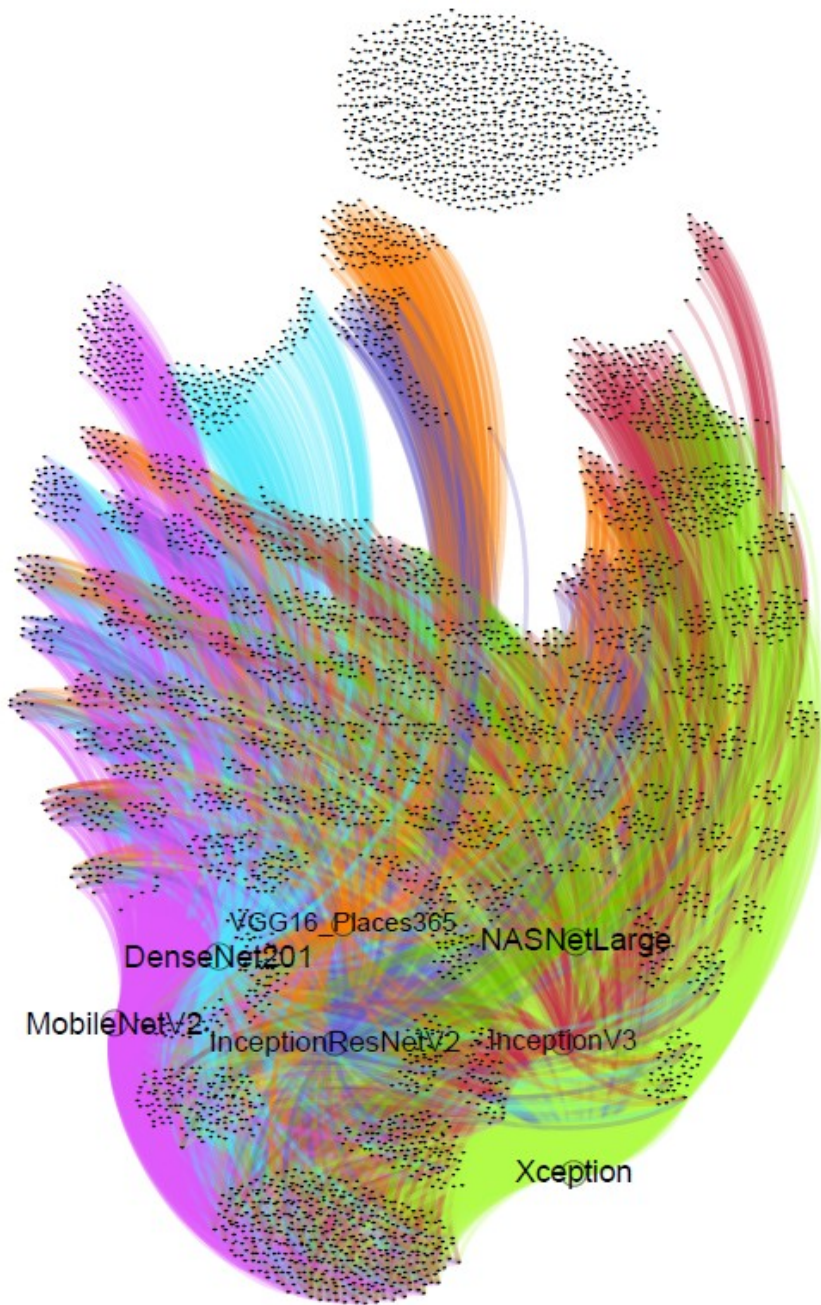


- Np. rekomendacja hashtagów do obrazów
- Zbiór referencyjny: HARRISON
- Minseok Park, Hanxiang Li, Junmo Kim, HARRISON: A Benchmark on Hashtag Recommendation for Real-world Images in Social Networks, arXiv:1605.05054

Problem klasyfikacji obiektu do kilku potencjalnych klas



- Przykładowy model rozwiązujący powyższy problem: VDNN-RN (Voting Deep Neural Network with Associative Rules Mining)
- Hachaj, T.; Miazga, J. Image Hashtag Recommendations Using a Voting Deep Neural Network and Associative Rules Mining Approach. *Entropy* 2020, 22, 1351. <https://doi.org/10.3390/e22121351>



Wizualizacja „pokrycia” zbioru danych przez poszczególne sieci. Krawędź grafu oznacza, że dana sieć prawidłowo określiła klasę, do której należy obrazek (test accuracy (5)). Kolory krawędzi są przypisane do poszczególnych sieci. Ważny wniosek – pretrenowane sieci nie działają jednakowo!

Sformułowanie problemu

- To each image I we want to assign a 4-dimensional real-valued vector with non-negative coordinates p , which approximates the potential voting of judges, using a certain approximation function Φ . Each dimension of the vector represents the number of votes that judges have cast for a certain class.

$$\Phi(I) = \bar{p}$$

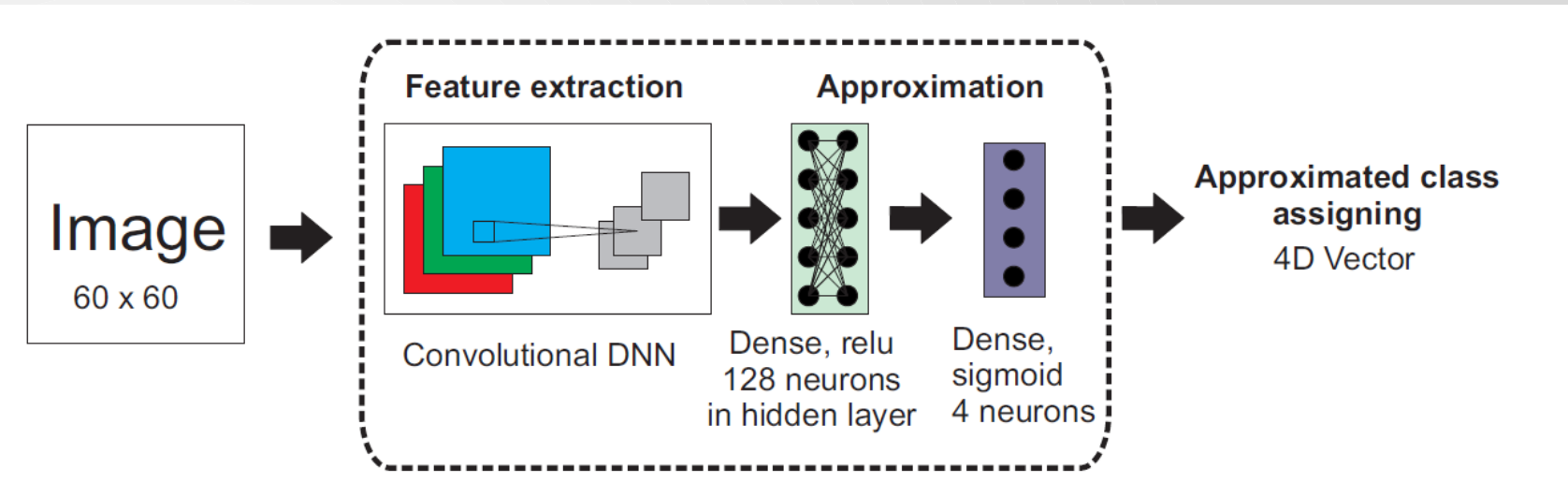
- To make the approximation independent of the number of judges that have taken part in data set preparation, we also assume that coordinates of vector p are scaled in range $[0, 1]$, where 0 means that no judge has voted for certain class while 1 that all judges have voted for it. We can easily transfer the votes of judges from vector v to n simply by division of each coordinate of v by number of judges k .

$$\bar{p} = \frac{\bar{v}}{k}$$

$$D = \{(I_i, \bar{p}_i); i = [0, \dots, n]\}$$

- Hachaj, T.; Bibrzycki, Ł.; Piekarczyk, M. Recognition of Cosmic Ray Images Obtained from CMOS Sensors Used in Mobile Phones by Approximation of Uncertain Class Assignment with Deep Convolutional Neural Network. Sensors 2021, 21, 1963. <https://doi.org/10.3390/s21061963>

Sformułowanie problemu



- Loss:

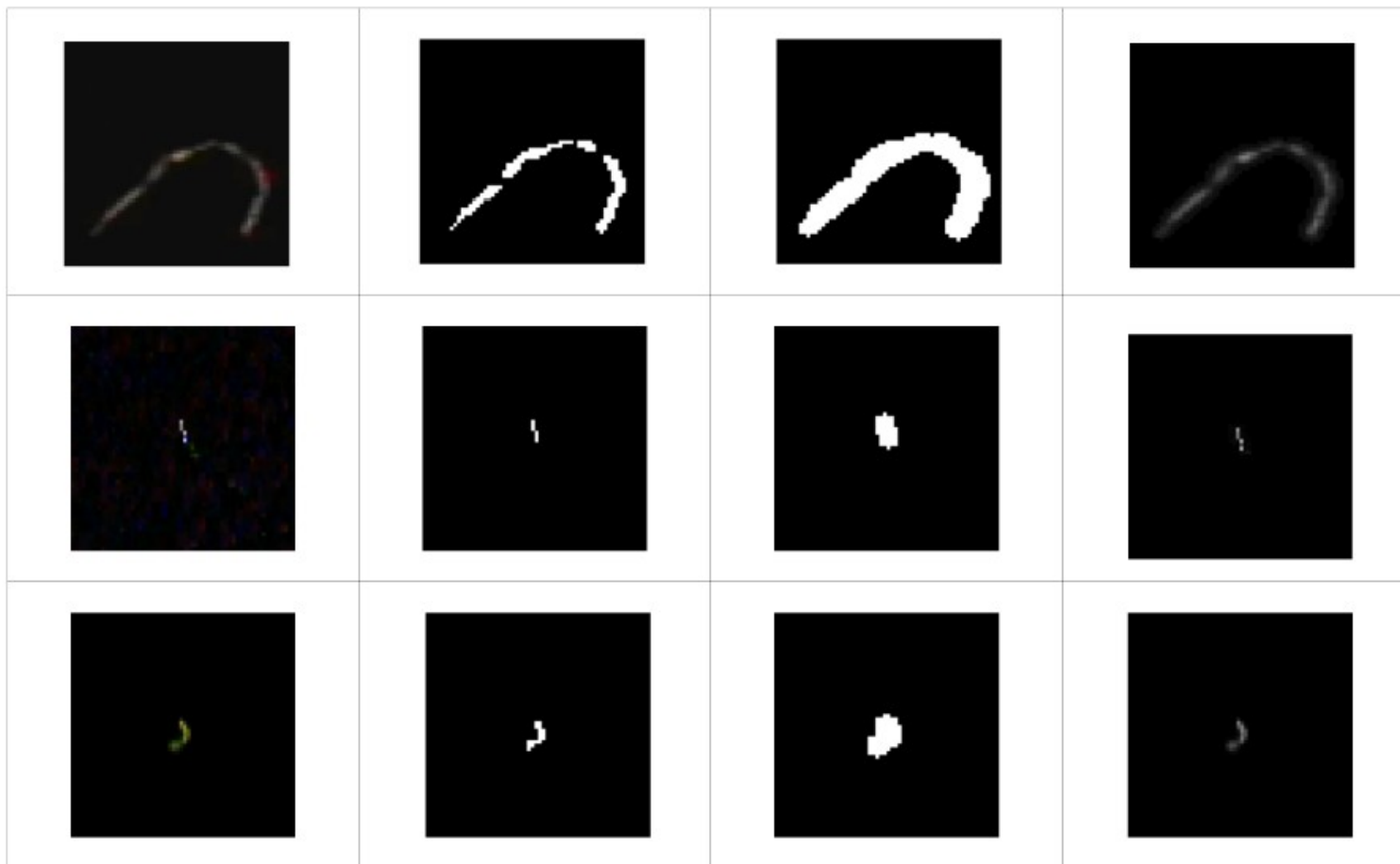
$$MSE = \frac{1}{n} \sum_0^n (\bar{p}_i - \bar{c}_i)^2$$

$$C_i = \max_{id} \bar{p}_i$$

$$C_i^t = \begin{cases} id & \text{if } \max_{id} \bar{p}_i > t \\ \emptyset & \text{if } \max_{id} \bar{p}_i \leq t \end{cases} \quad (8)$$

Wstępne przetwarzanie obrazów

(A) Input image (B) Otsu algorithm (C) Binary mask (D) Output image



Implementacja



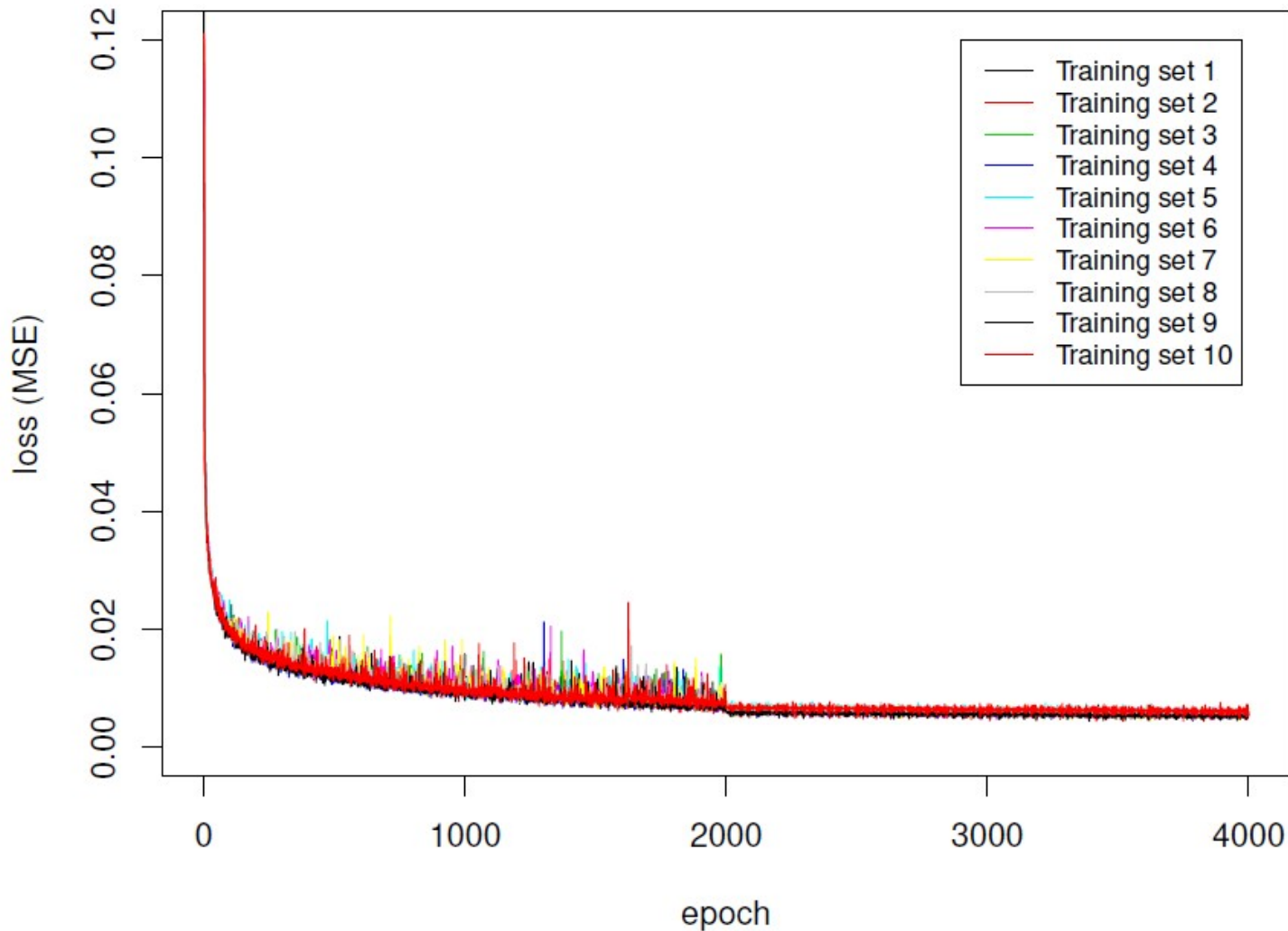
- Kody źródłowe: https://github.com/browarsoftware/credo_dnn_rays_recognition

Parametry treningu

- PC, Intel i7-9700F 3.00 Ghz CP, 64 GB RAM, NVIDIA GeForce RTX 2060 GPU, Windows 10 OS
- Optymalizator Adam (gradientowy)
- 4000 epok treningu
- 64 obiekty w batch-u
- Współczynnik uczenia 0.001 przez pierwsze 2000 epok, następnie 0.0001
- Dziesięciokrotna walidacja krzyżowa, 2115 losowych obiektów w zbiorze treningowym, 235 w walidacyjnym.

Wyniki

Input convolutional architecture	Recognition rate	Mean squared error
VGG16	85.79 ± 2.24	0.03 ± 0.00
NASNetLarge	81.66 ± 2.53	0.04 ± 0.01
MobileNetV2	78.43 ± 2.06	0.05 ± 0.01
Xception	81.49 ± 2.94	0.04 ± 0.01
DenseNet201	84.68 ± 1.77	0.03 ± 0.00



Wyniki

- VGG16

	Spots	Tracks	Worms	Artefacts
Spots	90.84±4.16	1.38±1.88	5.04±2.82	2.74±1.78
Tracks	7.46±3.80	73.31±11.50	15.10±9.01	4.14±2.53
Worms	6.14±6.01	22.64±10.70	62.59±9.92	8.64±4.19
Artefacts	2.54±1.23	0.80±1.05	2.70±1.60	93.97±1.93

- Xception

	Spots	Tracks	Worms	Artefacts
Spots	84.29±4.04	3.69±2.83	6.49±2.86	5.53±3.67
Tracks	8.22±5.66	60.21±7.30	23.58±7.44	7.99±4.66
Worms	7.58±4.77	21.18±5.23	60.18±8.70	11.06±4.65
Artefacts	2.01±1.21	1.40±1.09	3.14±1.71	93.45±2.18

- NASNetLarge

	Spots	Tracks	Worms	Artefacts
Spots	89.52±5.40	3.31±3.13	3.76±3.29	3.41±2.05
Tracks	9.27±4.13	62.26±5.06	17.96±4.93	10.52±5.46
Worms	7.83±4.16	26.41±9.08	51.66±6.54	14.11±5.30
Artefacts	2.44±1.79	1.58±0.39	2.71±1.50	93.27±2.71

- DenseNet201

	Spots	Tracks	Worms	Artefacts
Spots	87.44±4.97	4.79±2.74	4.25±2.61	3.52±2.54
Tracks	5.11±3.06	71.03±5.43	20.33±5.41	3.53±3.49
Worms	6.03±5.00	23.46±7.71	61.90±10.16	8.61±3.15
Artefacts	2.63±0.86	0.79±0.87	1.93±1.17	94.65±1.86

- MobileNetV2

	Spots	Tracks	Worms	Artefacts
Spots	78.66±3.46	13.37±4.03	4.34±2.70	3.63±2.34
Tracks	12.28±6.03	56.14±5.78	24.40±4.50	7.18±5.96
Worms	7.15±5.44	25.63±9.40	50.84±7.89	16.39±3.87
Artefacts	2.09±1.29	1.22±0.61	3.15±2.04	93.54±1.74

Wyniki

Table 9: Confusion matrix of network with input VGG16 and threshold schema (equation 8) $t=0.75$. Data remains = $74.26\% \pm 2.56$ (D $63.12\% \pm 4.63$; L $55.60\% \pm 5.83$; W $34.71\% \pm 7.93$; A $96.46\% \pm 2.08$), recognition rate = $94.95\% \pm 1.00$.

	Spots (S)	Tracks (T)	Worms (W)	Artefacts (A)
Spots	96.23 ± 2.29	0.92 ± 1.49	0.36 ± 1.13	2.49 ± 2.38
Tracks	0.83 ± 2.64	88.96 ± 6.79	8.72 ± 5.25	1.49 ± 3.46
Worms	1.00 ± 3.16	14.24 ± 11.74	79.54 ± 10.97	5.22 ± 5.75
Artefacts	0.65 ± 0.64	0.56 ± 0.48	1.80 ± 0.81	97.00 ± 0.95

Table 10: Confusion matrix of network with input VGG16 and threshold schema (equation 8) $t=0.90$. Data remains = $62.60\% \pm 2.88$ (D $42.03\% \pm 5.65$; L $29.91\% \pm 6.55$; W $18.45\% \pm 7.22$; A $94.97\% \pm 2.17$), recognition rate = $96.95\% \pm 1.38$.

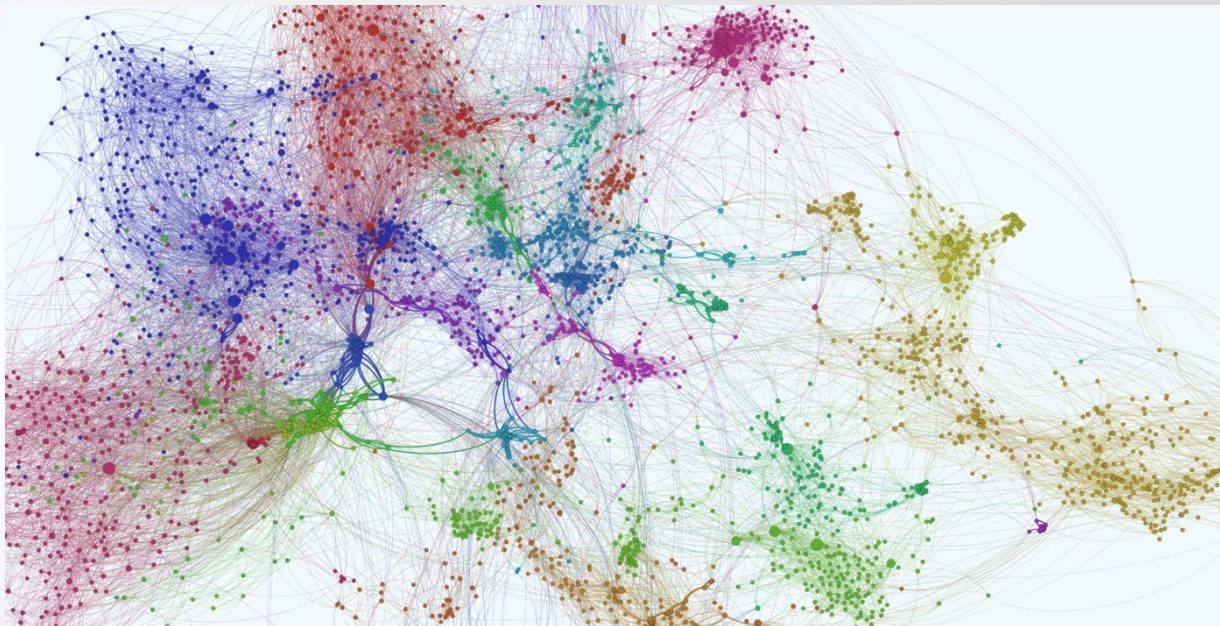
	Spots (S)	Tracks (T)	Worms (W)	Artefacts (A)
Spots	98.71 ± 2.99	0.45 ± 1.44	0.00 ± 0.00	0.84 ± 1.78
Tracks	1.43 ± 4.52	88.89 ± 12.78	8.85 ± 8.78	0.83 ± 2.64
Worms	0.00 ± 0.00	7.43 ± 15.82	89.65 ± 15.52	2.92 ± 6.23
Artefacts	$.38 \pm 0.50$	0.27 ± 0.44	1.64 ± 0.81	97.70 ± 1.24

Wnioski

- Model sieci działa zgodnie z oczekiwaniami, przy jego pomocy można aproksymować proce głosowania „sędziów”.
Manipulując współczynnikiem progowym możemy pomijać obiekty, których przypisanie do klasy nie jest wystarczająco pewne.
- Prawdopodobnie konwolucyjny ekstraktor cech sieci VGG16 ma nadmiarową architekturę – można uprościć.
- Uproszczony ekstraktor cech po wytrenowaniu można użyć do klasteryzacja zbioru, dzięki czemu możliwe będzie np. wyszukiwanie podobnych morfologicznie obrazów.

Dziękuję za uwagę!

- Bardzo proszę o pytania 😊



- Hachaj, T.; Bibrzycki, Ł.; Piekarczyk, M. Recognition of Cosmic Ray Images Obtained from CMOS Sensors Used in Mobile Phones by Approximation of Uncertain Class Assignment with Deep Convolutional Neural Network. Sensors 2021, 21, 1963. <https://doi.org/10.3390/s21061963>
- Kody źródłowe: https://github.com/browarsoftware/credo_dnn_rays_recognition