



INPAINTING

Weronika Hryniewska

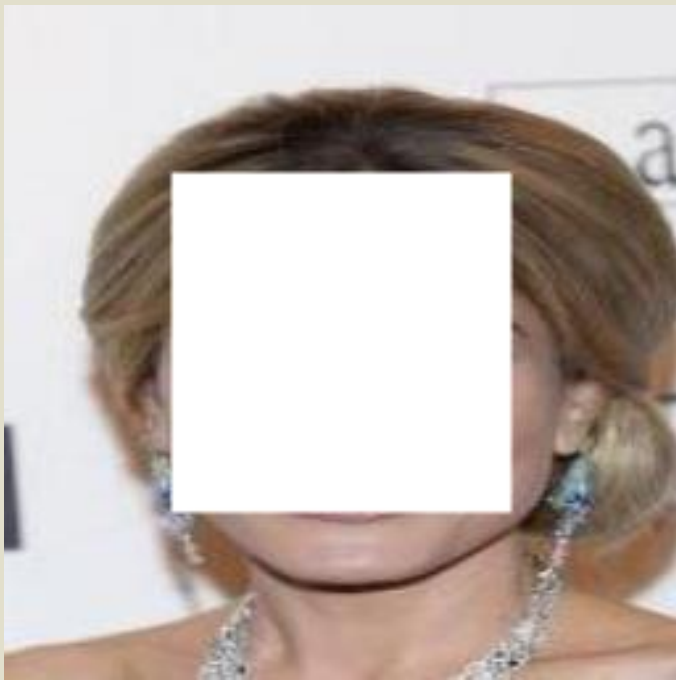
Zastosowania

- Usuwanie niechcianych fragmentów obrazu
- Odrestaturowywanie starych zdjęć lub filmów
- Usuwanie znaków wodnych
- Usuwanie log
- Efekty kreatywne



Rodzaje ze wzg. na rodzaj masek

- Prostokątne

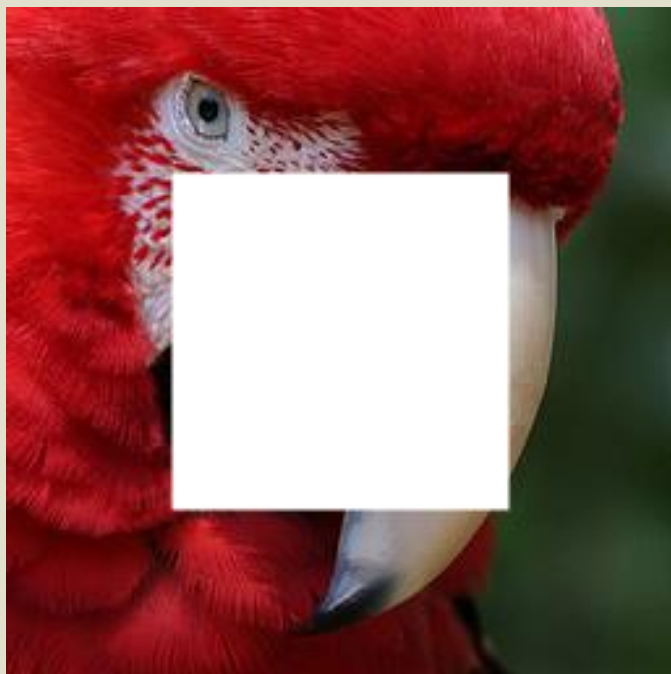


- Nieregularne



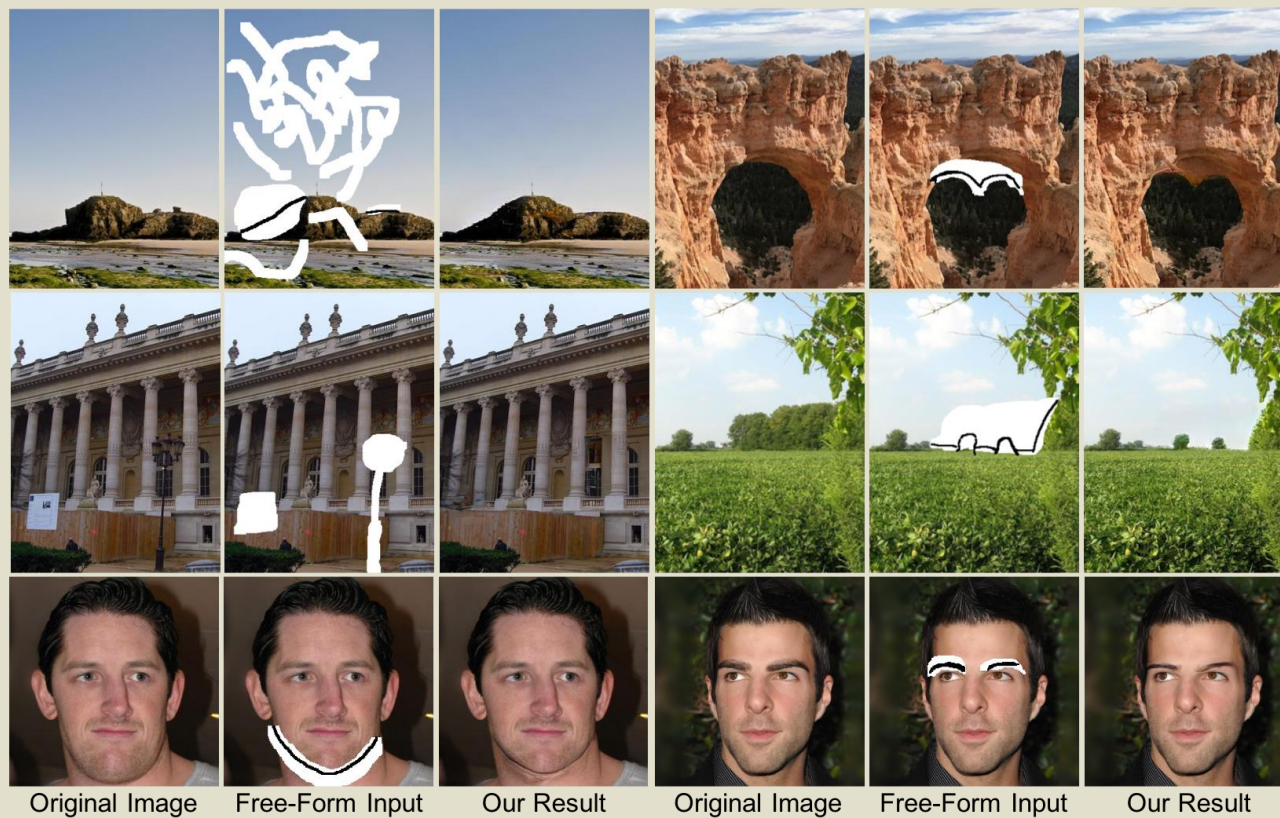
Rodzaje ze wzg. na liczbę masek

◦ jedna



◦ wiele





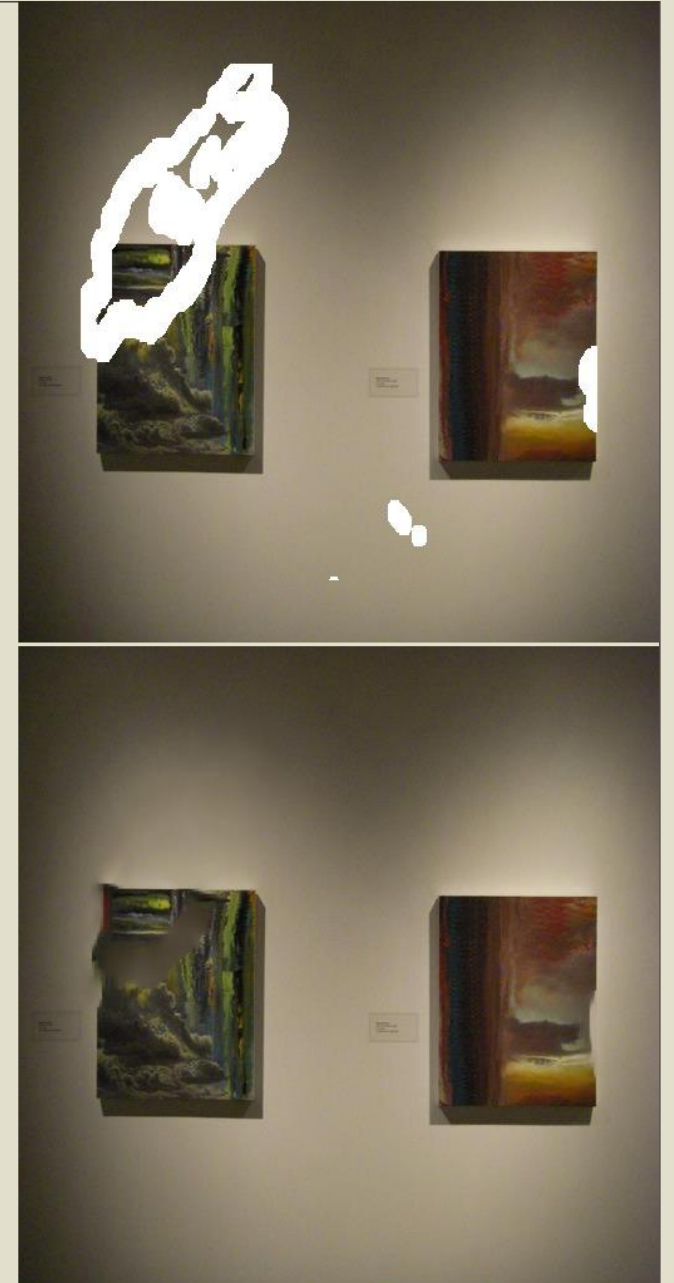
[1] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu and T. Huang, ‘Free-Form Image Inpainting with Gated Convolution’, *arXiv:1806.03589 [cs]*, Oct 2019.



SPOSOBY REALIZACJI INPAINTINGU

PatchMatch

- Statystyka pozostałej części obrazu
- PatchMatch – iteracyjnie szuka najlepiej pasujących łatek, aby wpasować je w dziury
- Rezultaty są gładkie
- Ograniczenia:
 - obszar obrazu dostępny do wykonania statystyki
 - brak „rozumienia” obrazu



Deep learning

- uczą się semantyki, znaczenia obrazu w sposób bardziej całościowy
- Stosowane są filtry konwolucyjne, które zamieniają usunięty fragment obrazu stałą wartością
- Ograniczenia:
 - Zależność od początkowej, ustalonej wartości dziur
 - Problem z tekstuowaniem regionów dziur
 - Problem z oczywistym kontrastem pomiędzy kolorami
 - Sztuczność krawędzi na styku dziury i obrazu



Postprocessing

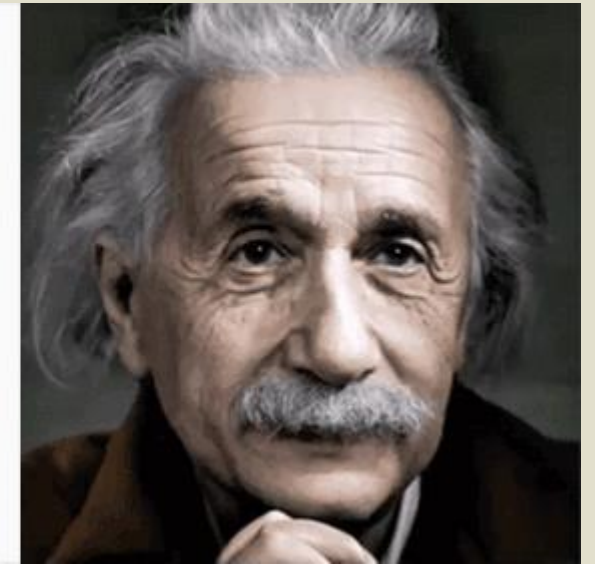
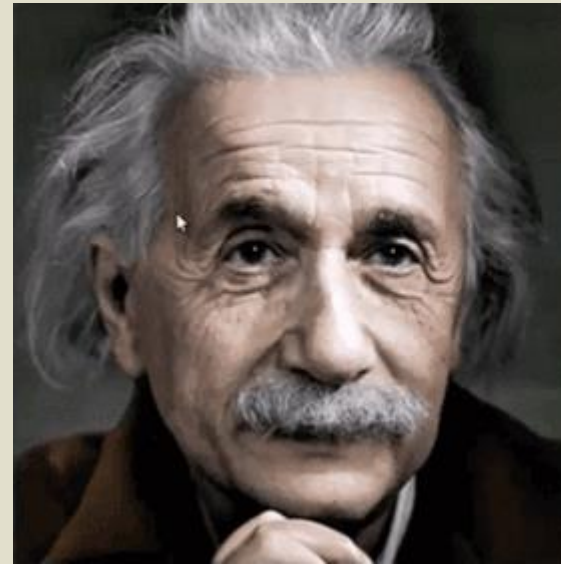
- Fast marching
- Poisson image blending
- Sieć czyszcząca, która oczyszcza surowe predykcje sieci





ROZWIĄZANIE NVIDIA

Image Inpainting for Irregular Holes Using Partial Convolutions



[1] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, 'Image Inpainting for Irregular Holes Using Partial Convolutions', *arXiv:1804.07723 [cs]*, Apr. 2018.

Do tej pory:

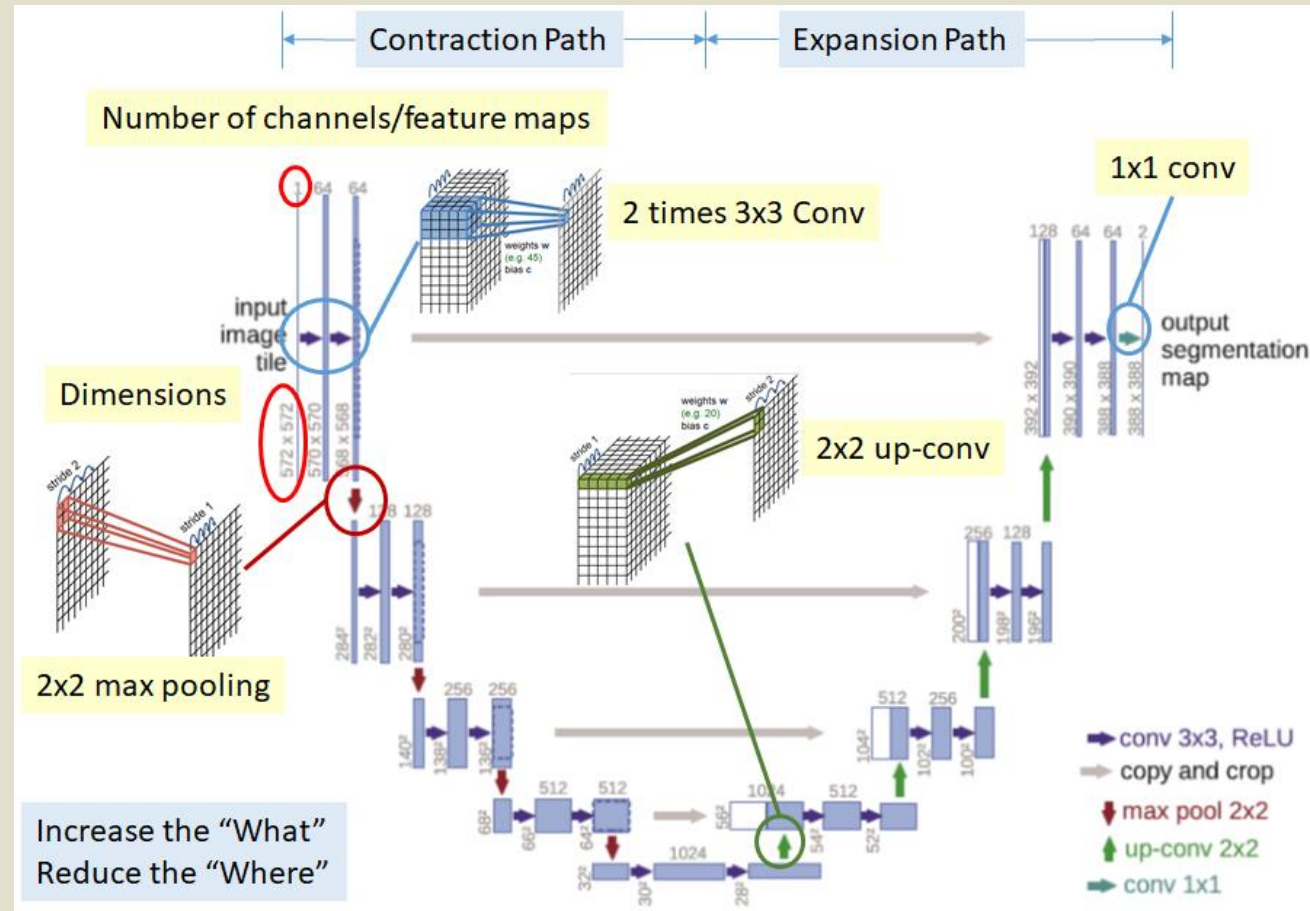
- standardowe sieci konwolucyjne używane na prawidłowych pikselach jak i na dziurach pokrytych maską (zazwyczaj średnią)
- Efekt: niezgodności na obrazie i rozmycie
- Konieczny postprocessing, który usunie artefakty

Teraz:

- Częściowe konwolucje, gdzie konwolucje działają tylko na prawidłowych pikselach - pokrycie maską i znormalizowanie
- Mechanizm do automatycznej generacji i aktualizacji masek

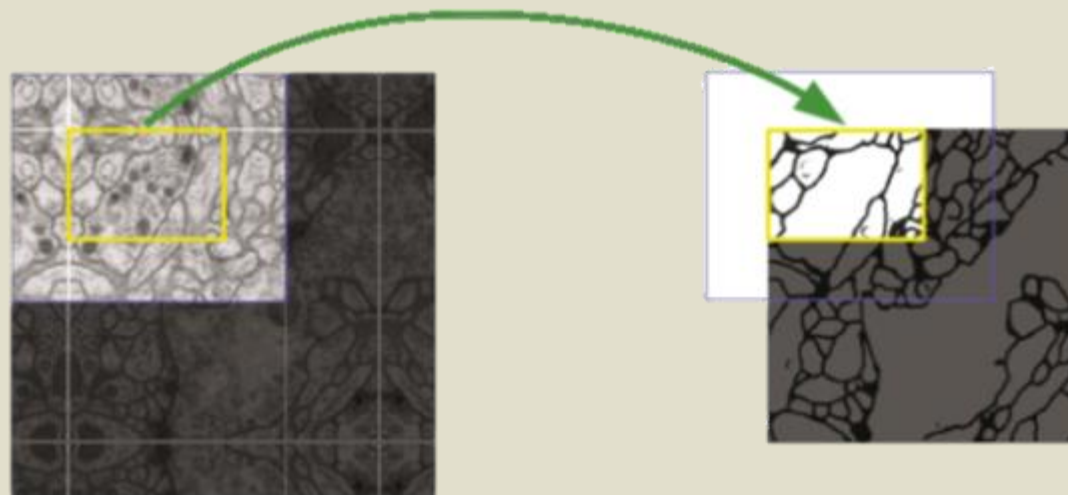
Architektura sieci i implementacja

- Rozszerzenie PyTorch'a o warstwę częściowych konwolucji
- Maski binarne o wymiarach takich samych jak powiązane z nimi obrazy ($C \times H \times W$)
- Aktualizacja mask zaimplementowana poprzez użycie stałej warstwy konwolucyjnej o takiej samej wielkości, co operacja częściowych konwolucji, ale o wagach ustawionych na 1 lub bez biasu
- Czas wykonania inpaintingu jednego obrazka 512×512 pikseli to 0.029s
- Trenowanie na pojedynczej karcie NVIDIA V100

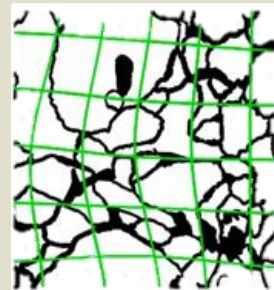
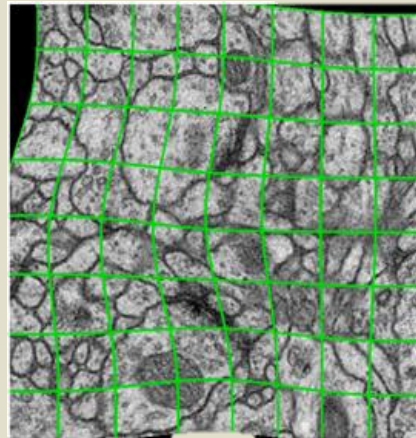


O. Ronneberger, P. Fischer, and T. Brox, 'U-Net: Convolutional Networks for Biomedical Image Segmentation', *arXiv:1505.04597 [cs]*, May 2015.

- Strategia nakładających się płytek



- Elastyczne deformacje w celu zwiększenia danych treningowych



Zaproponowana metoda

- Warstwa częściowych konwolucji

$$x' = \begin{cases} \mathbf{W}^T (\mathbf{X} \odot \mathbf{M}) \frac{\text{sum}(\mathbf{1})}{\text{sum}(\mathbf{M})} + b, & \text{if } \text{sum}(\mathbf{M}) > 0 \\ 0, & \text{otherwise} \end{cases}$$

- Aktualizacja maski

$$m' = \begin{cases} 1, & \text{if } \text{sum}(\mathbf{M}) > 0 \\ 0, & \text{otherwise} \end{cases}$$

Funkcja strat

$$\mathcal{L}_{total} = \mathcal{L}_{valid} + 6\mathcal{L}_{hole} + 0.05\mathcal{L}_{perceptual} + 120(\mathcal{L}_{style_{out}} + \mathcal{L}_{style_{comp}}) + 0.1\mathcal{L}_{tv}$$

$$\mathcal{L}_{valid} = \frac{1}{N_{\mathbf{I}_{gt}}} \|M \odot (\mathbf{I}_{out} - \mathbf{I}_{gt})\|_1$$

$$\mathcal{L}_{hole} = \frac{1}{N_{\mathbf{I}_{gt}}} \|(1 - M) \odot (\mathbf{I}_{out} - \mathbf{I}_{gt})\|_1$$

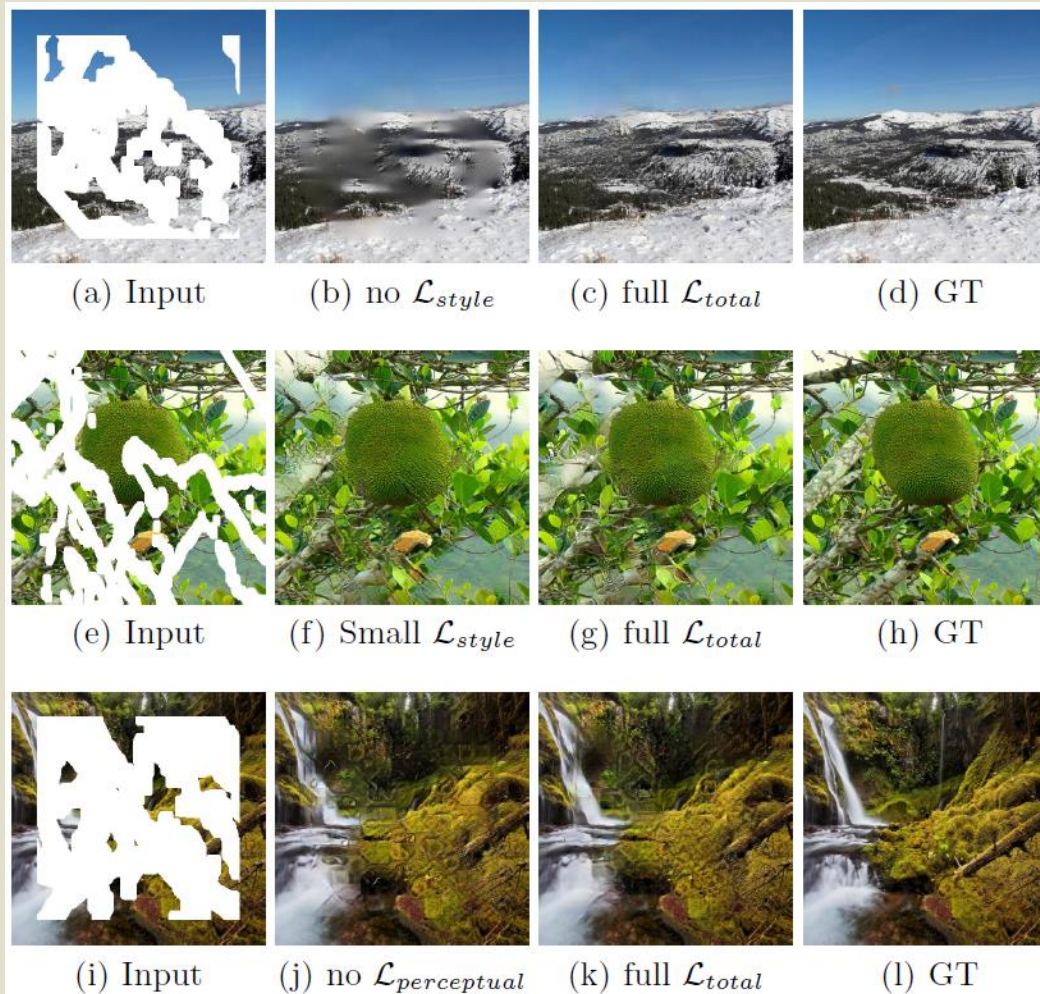
$$\mathcal{L}_{perceptual} = \sum_{p=0}^{P-1} \frac{\|\Psi_p^{\mathbf{I}_{out}} - \Psi_p^{\mathbf{I}_{gt}}\|_1}{N_{\Psi_p^{\mathbf{I}_{gt}}}} + \sum_{p=0}^{P-1} \frac{\|\Psi_p^{\mathbf{I}_{comp}} - \Psi_p^{\mathbf{I}_{gt}}\|_1}{N_{\Psi_p^{\mathbf{I}_{gt}}}} \quad [2]$$

$$\mathcal{L}_{style_{out}} = \sum_{p=0}^{P-1} \frac{1}{C_p C_p} \left\| K_p((\Psi_p^{\mathbf{I}_{out}})^\top (\Psi_p^{\mathbf{I}_{out}}) - (\Psi_p^{\mathbf{I}_{gt}})^\top (\Psi_p^{\mathbf{I}_{gt}})) \right\|_1$$

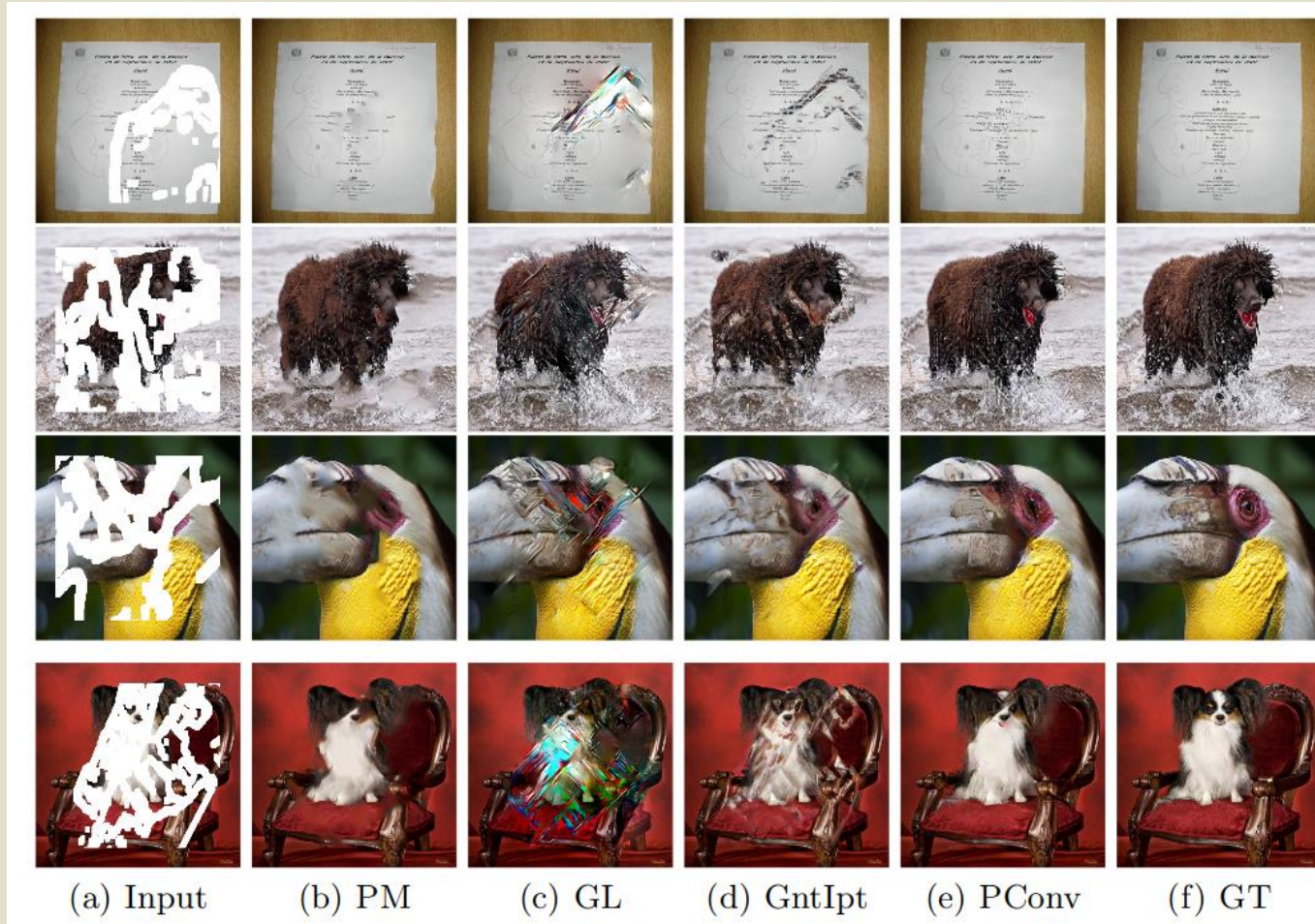
$$\mathcal{L}_{style_{comp}} = \sum_{p=0}^{P-1} \frac{1}{C_p C_p} \left\| K_p((\Psi_p^{\mathbf{I}_{comp}})^\top (\Psi_p^{\mathbf{I}_{comp}}) - (\Psi_p^{\mathbf{I}_{gt}})^\top (\Psi_p^{\mathbf{I}_{gt}})) \right\|_1$$

$$\mathcal{L}_{tv} = \sum_{(i,j) \in R, (i,j+1) \in R} \frac{\|\mathbf{I}_{comp}^{i,j+1} - \mathbf{I}_{comp}^{i,j}\|_1}{N_{\mathbf{I}_{comp}}} + \sum_{(i,j) \in R, (i+1,j) \in R} \frac{\|\mathbf{I}_{comp}^{i+1,j} - \mathbf{I}_{comp}^{i,j}\|_1}{N_{\mathbf{I}_{comp}}}$$

$$\mathcal{L}_{total} = \mathcal{L}_{valid} + 6\mathcal{L}_{hole} + 0.05\mathcal{L}_{perceptual} + 120(\mathcal{L}_{style_{out}} + \mathcal{L}_{style_{comp}}) + 0.1\mathcal{L}_{tv}$$



Wyniki na zbiorze obrazków “ImageNet”



Dziękuję za uwagę!