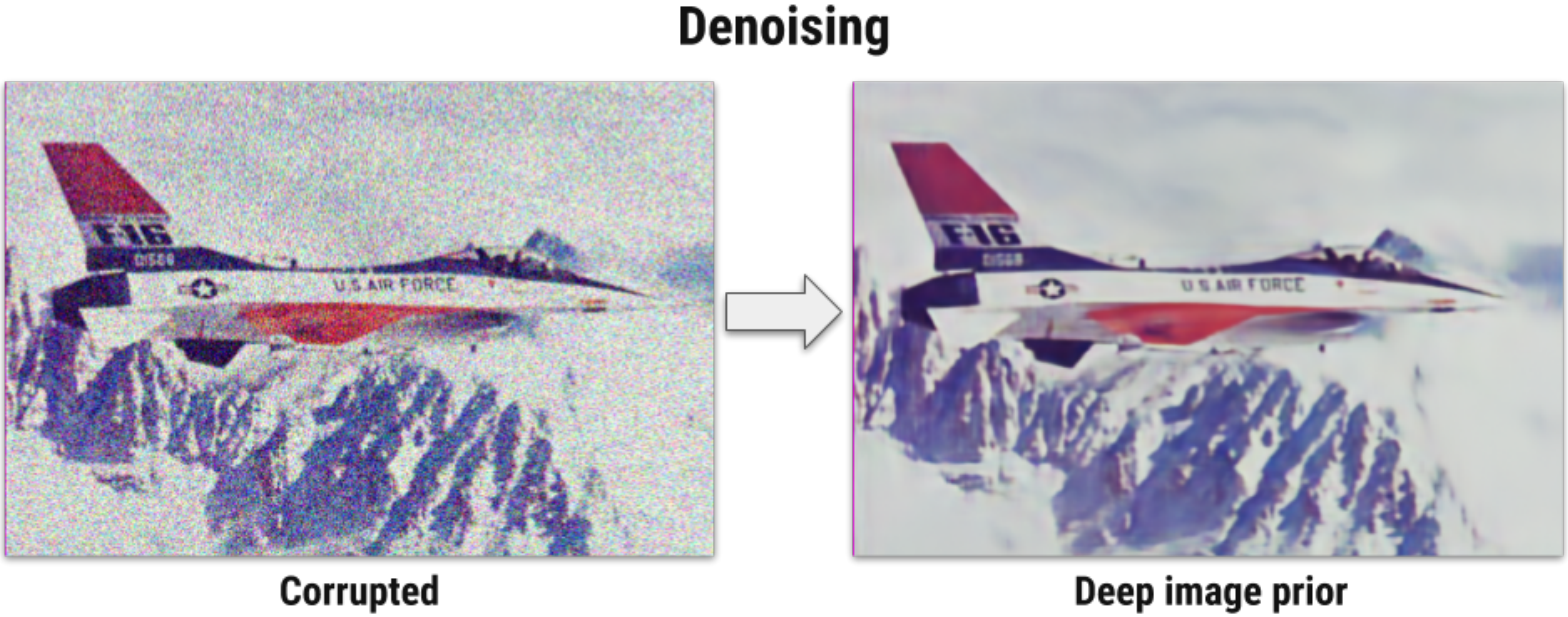
Deep Image Prior

Кнунянц Иван БПМИ171

Задачи image restoration



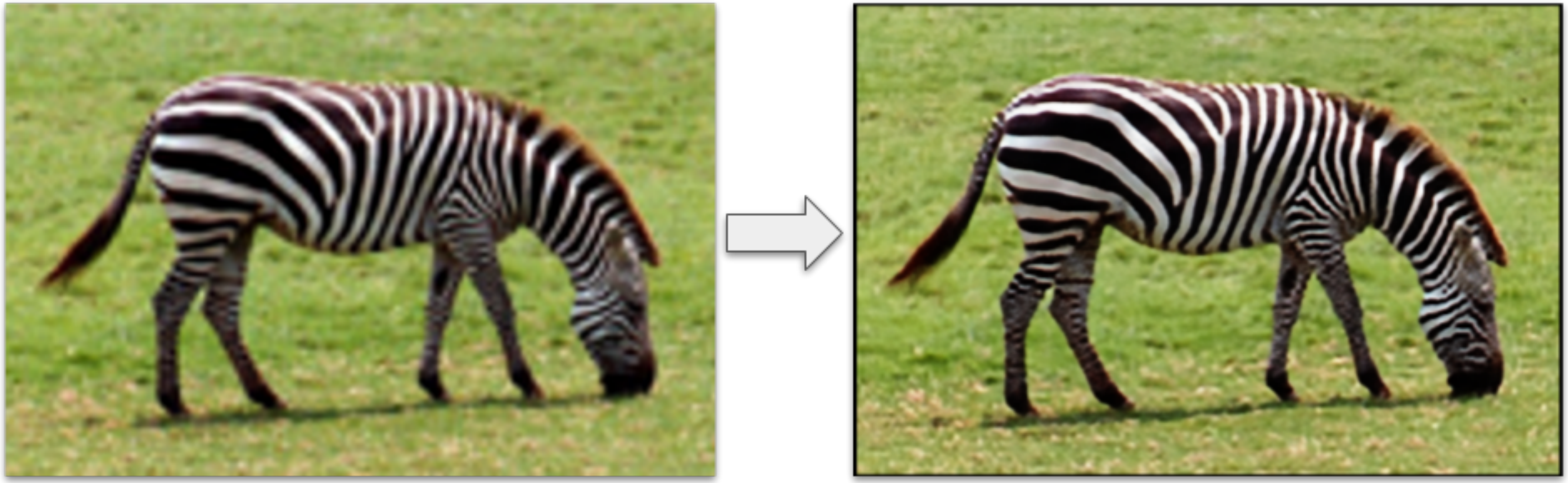
Задачи image restoration

Inpainting

Corrupted — Deep image prior

Задачи image restoration

Super-resolution



Corrupted

Deep image prior

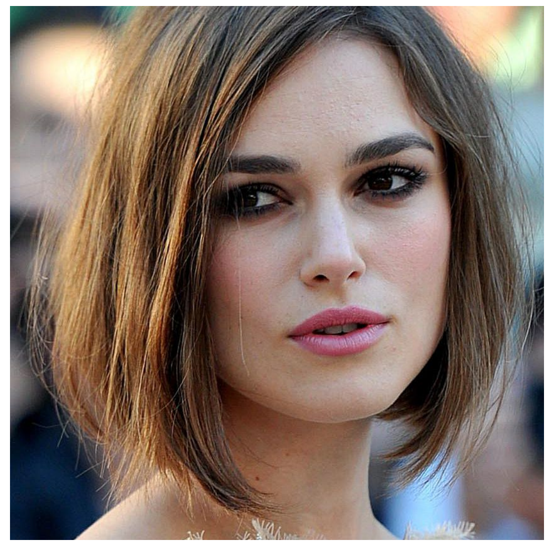
Что такое Prior?

Prior - это наше “знание о мире”, которое выучивает нейросеть.

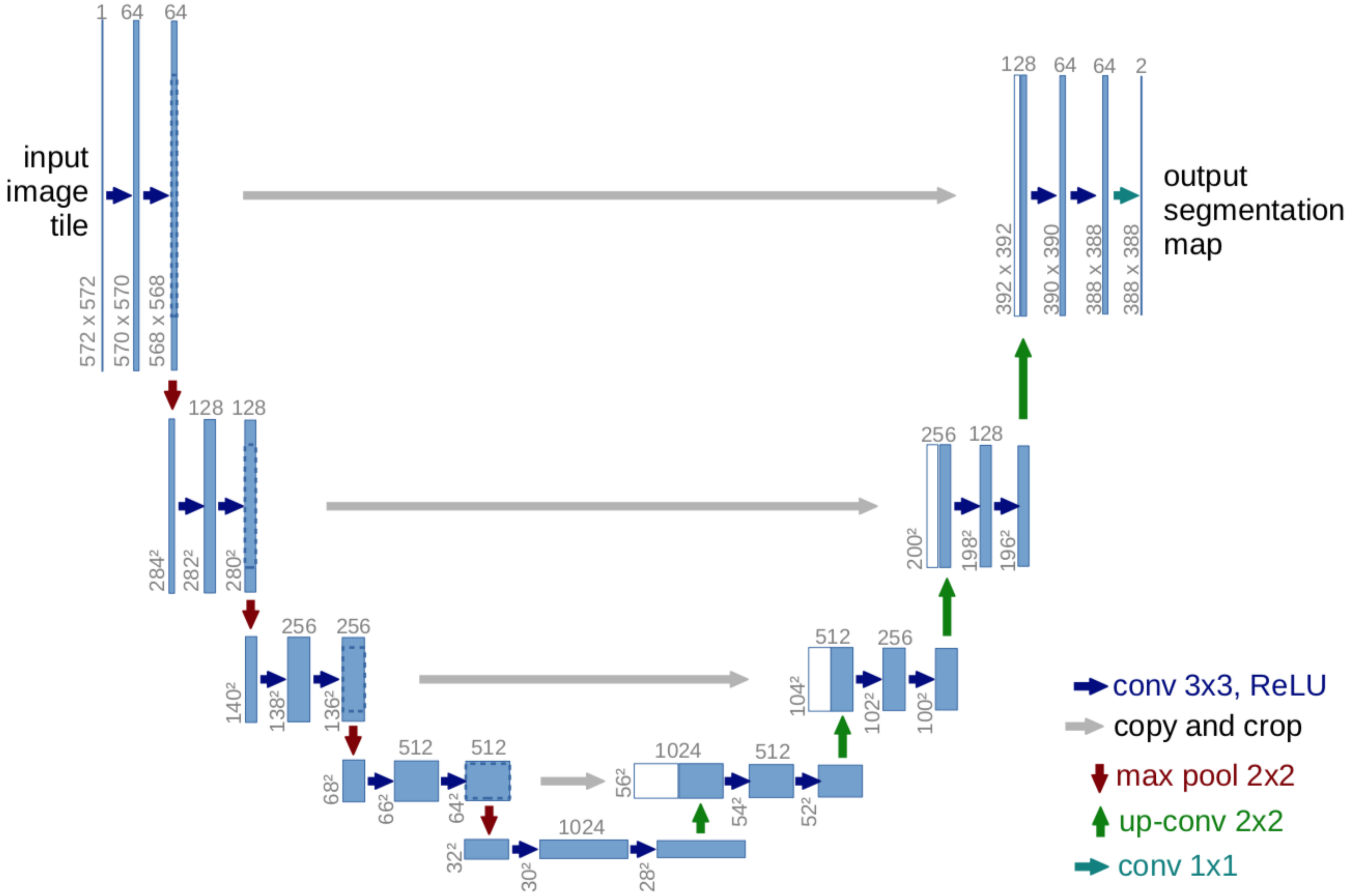
Составляющие prior’a

Выученный prior

→



Структура нейросети



Результаты

(a) Ground truth
(b) SRResNet [19], **Trained**
(c) Bicubic, **Not trained**
(d) Deep prior, **Not trained**

Идея image restoration

x - искомое изображение

$\hat{x}$ - испорченное изображение

Полученное изображение: $x^* = \arg\max_x p(x \mid \hat{x})$

$$p(x \mid \hat{x}) = \frac{p(\hat{x} \mid x) p(x)}{p(\hat{x})} \Rightarrow p(\hat{x} \mid x) p(x) \to maximize$$

$p(\hat{x} \mid x)$ - правдоподобие, $p(x)$ - prior

Идея image restoration

Запишем выражение по-другому:

$$x^* = \arg\max_x p(x|\hat{x}) = \arg\max_x p(\hat{x}|x)p(x) =$$

$$= \arg\min_x - \log(p(\hat{x}|x)) - \log(p(x)) = \arg\min_x E(x;\hat{x}) + R(x)$$

$E(x;\hat{x})$ - data term, $R(x)$ - image prior

Параметризация

Пусть $g : \theta \mapsto x$ сюръективна. Тогда задача эквивалентна:

$$\arg\min_\theta E(g(\theta); \hat{x}) + R(g(\theta))$$

Теперь будем настраивать g, который даёт лучший результат.

Будем использовать g в качестве prior. Тогда будем оптимизировать только $\arg\min_\theta E(g(\theta); \hat{x})$

Зафиксируем $g \equiv f_\theta(z)$, где z - фиксированная величина (картинка), f_θ - свёрточная сеть с параметрами θ

Алгоритм

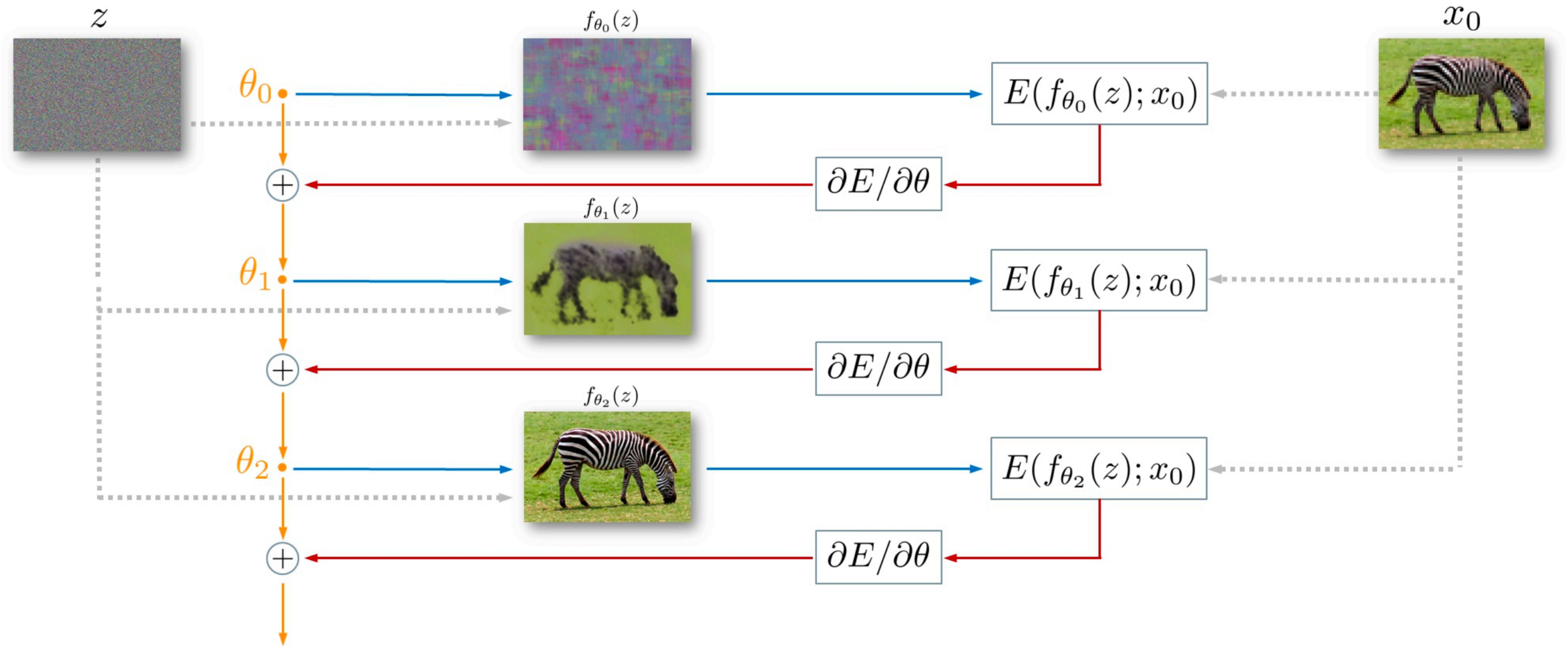
1. Инициализируем z. Например, равномерным шумом $U(-1,1)$.

2. Решаем $\arg\min_\theta E(f_\theta(z); \hat{x})$

$$\theta^{k+1} = \theta^k - \alpha \frac{\partial E(f_\theta(z); \hat{x})}{\partial \theta}$$

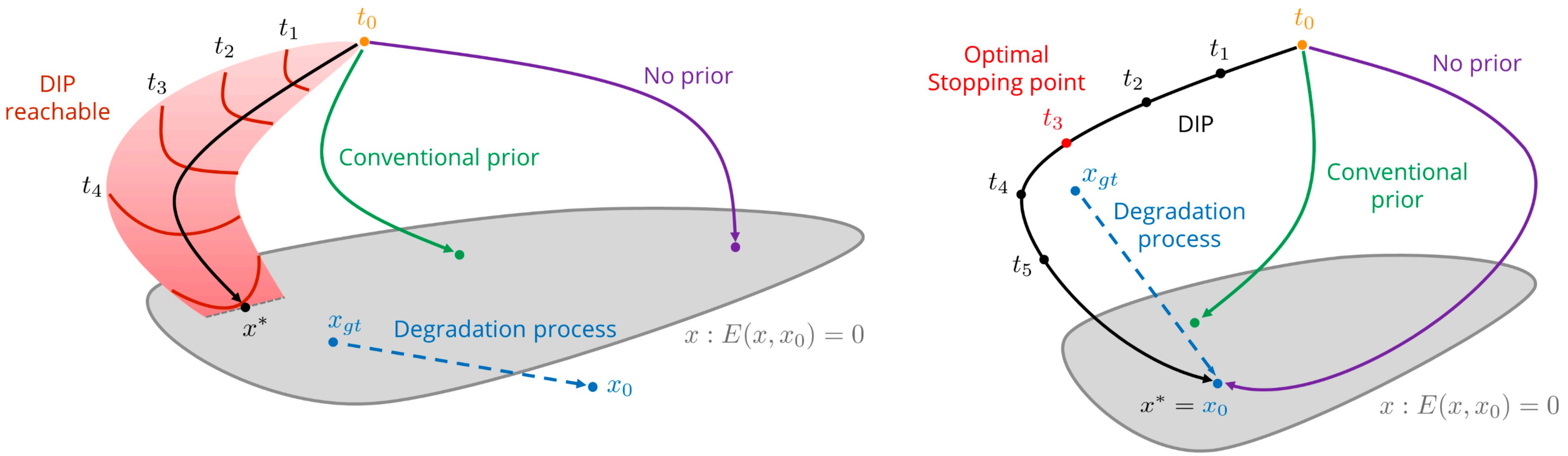
3. Получаем решение $x* = f_{\theta*}(z)$

Алгоритм



z
$f_{\theta_0}(z)$
x_0
θ_0
$E(f_{\theta_0}(z); x_0)$
$\partial E / \partial \theta$
$f_{\theta_1}(z)$
θ_1
$E(f_{\theta_1}(z); x_0)$
$\partial E / \partial \theta$
$f_{\theta_2}(z)$
θ_2
$E(f_{\theta_2}(z); x_0)$
$\partial E / \partial \theta$

Как работает prior в Deep Image Prior



Архитектура

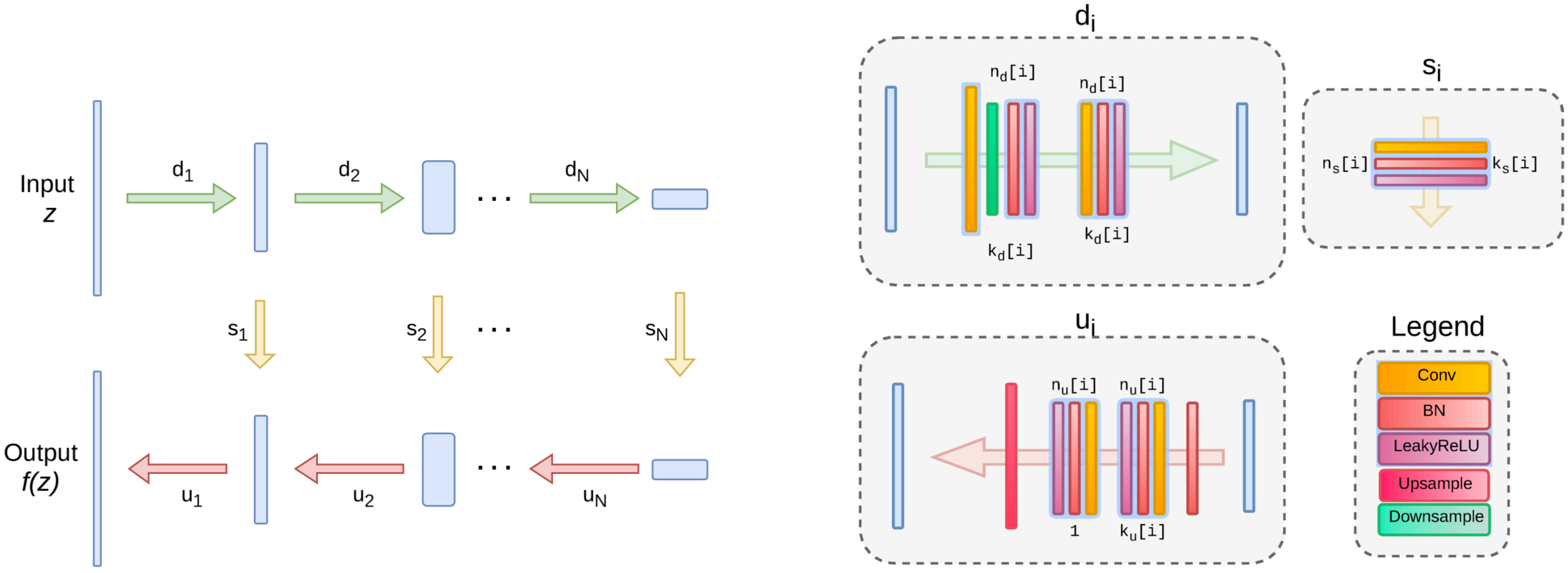


Fig. 21: **The architecture used in the experiments.** We use "hourglass" (also known as "decoder-encoder") architecture. We sometimes add skip connections (yellow arrows). $n_u[i]$, $n_d[i]$, $n_s[i]$ correspond to the number of filters at depth i for the upsampling, downsampling and skip-connections respectively. The values $k_u[i]$, $k_d[i]$, $k_s[i]$ correspond to the respective kernel sizes.

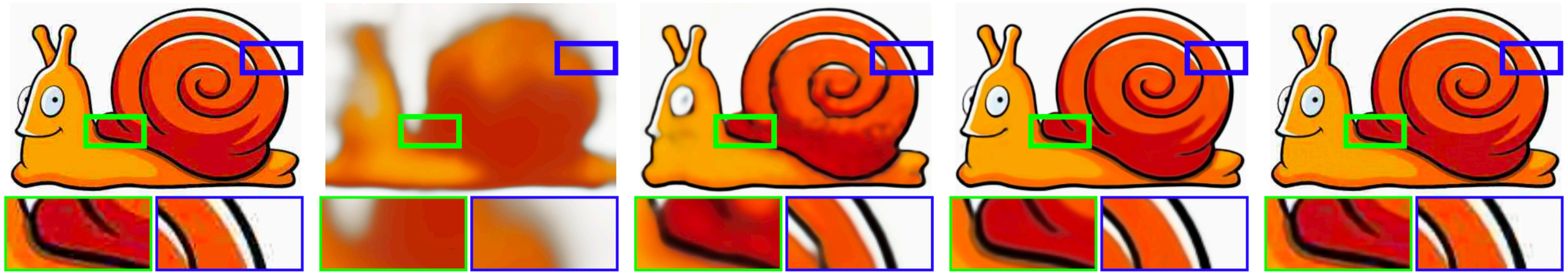
Data terms

- **Denoising:** $E(x; \hat{x}) = \|x - \hat{x}\|^2$

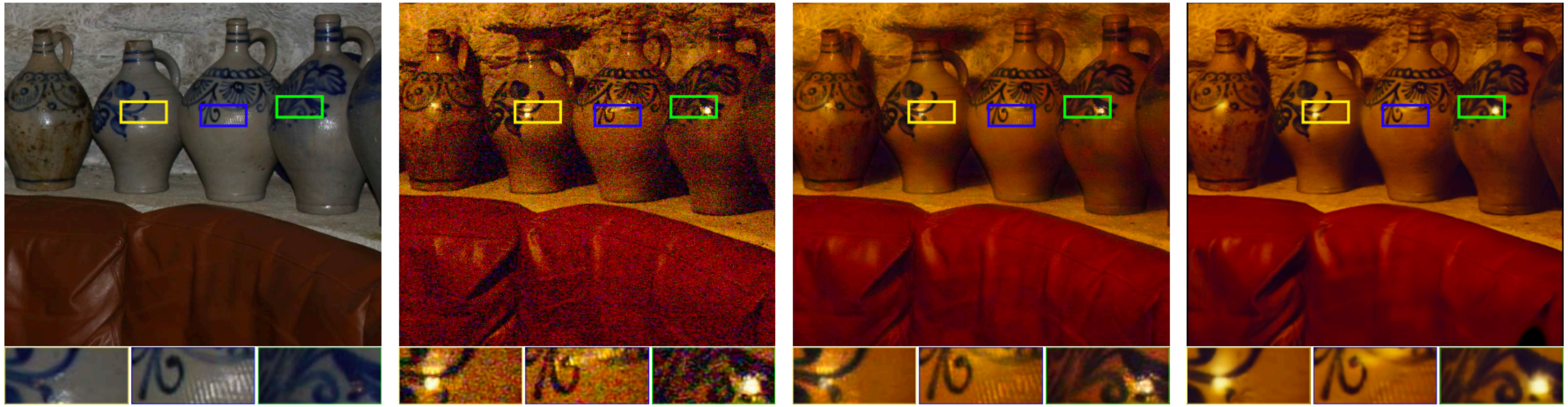
- **Inpainting:** $E(x; \hat{x}) = \|(x - \hat{x}) \cdot m\|^2$, m - маска

- **Super-resolution:** $E(x; \hat{x}) = \|d(x) - \hat{x}\|^2$

Denoising



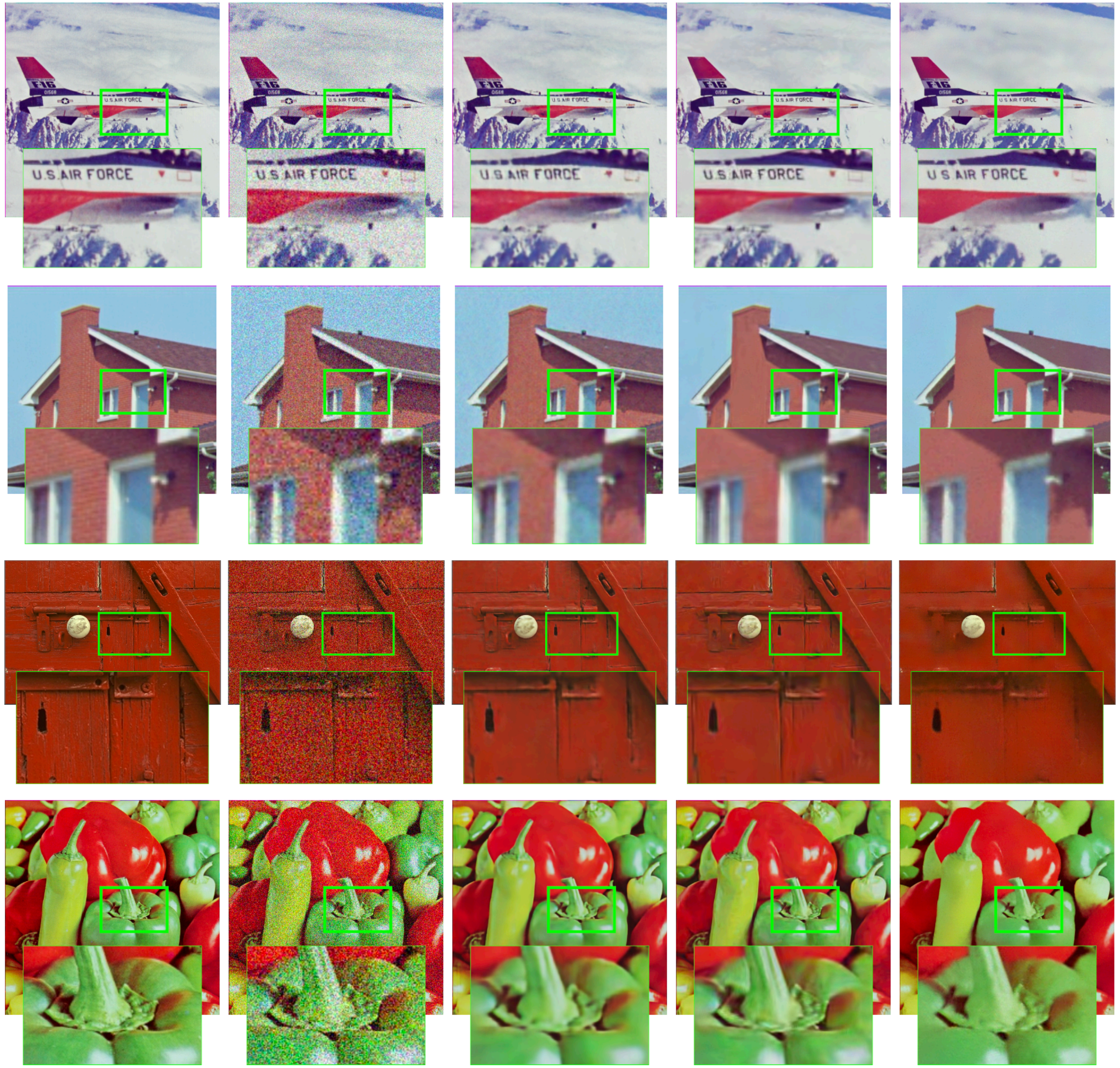
Corrupted 100 iterations 600 iterations 2400 iterations 50K iterations



(a) Flash (b) No flash (c) Joint bilateral [43] (d) Deep image prior

Fig. 20: **Reconstruction based on flash and no-flash image pair.** The deep image prior allows to obtain low-noise reconstruction with the lighting very close to the no-flash image. It is more successful at avoiding "leaks" of the lighting patterns from the flash pair than joint bilateral filtering [43] (c.f. blue inset).

Denoising



(a) GT (b) Input (c) Ours (d) CBM3D (e) NLM

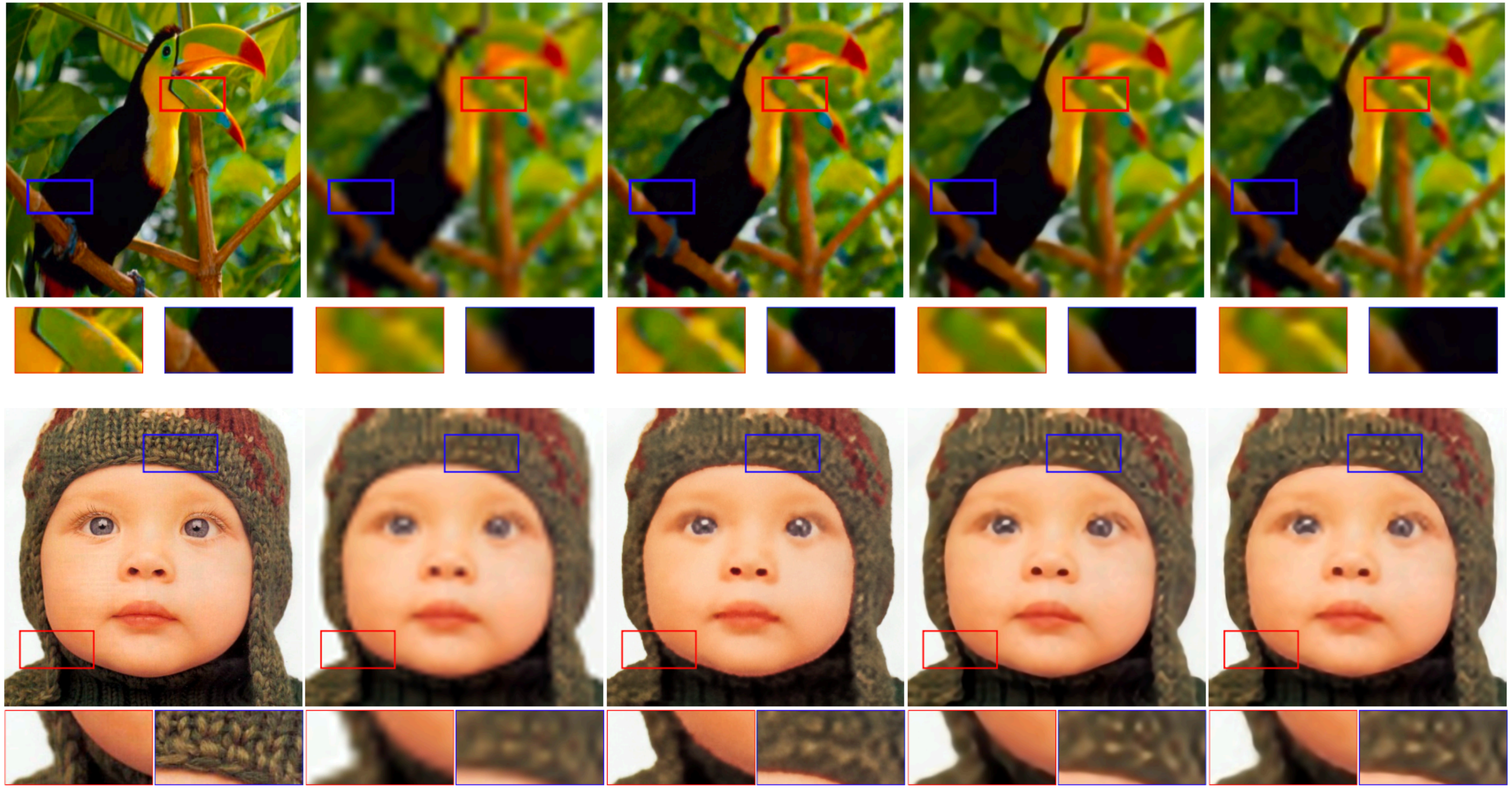
Super-resolution

4× super-resolution

(a) Original image (b) Bicubic, **Not trained** (c) Ours, **Not trained** (d) LapSRN, **Trained** (e) SRResNet, **Trained**

Super-resolution

8× super-resolution



(f) Original image (g) Bicubic, **Not trained** (h) Ours, **Not trained** (i) LapSRN, **Trained** (j) VDSR, **Trained**

Inpainting

(a) Input (white=masked)

(b) Encoder-decoder, depth=6

(c) Encoder-decoder, depth=4

(d) Encoder-decoder, depth=2

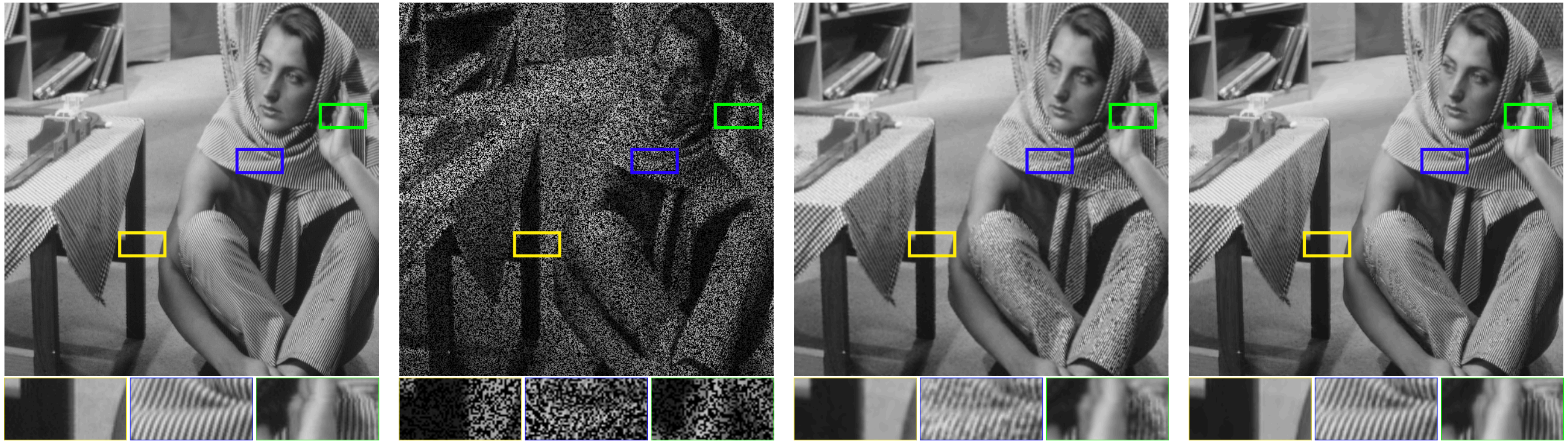
(e) ResNet, depth=8

(f) U-net, depth=5

Inpainting

	Barbara	Boat	House	Lena	Peppers	C.man	Couple	Finger	Hill	Man	Montage
Papyan et al.	28.14	31.44	34.58	35.04	31.11	27.90	31.18	31.34	32.35	31.92	28.05
Ours	**32.22**	**33.06**	**39.16**	**36.16**	**33.05**	**29.8**	**32.52**	**32.84**	**32.77**	**32.20**	**34.54**

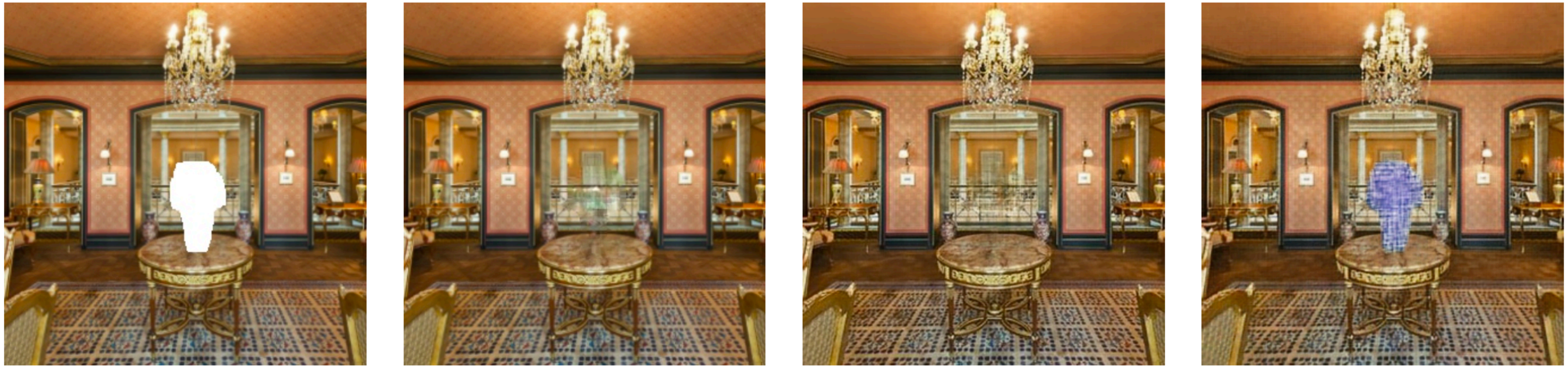
Fig. 12: Comparison between our method and the algorithm in [42]. See fig. 13 for visual comparison.



(a) Original image (b) Corrupted image (c) CSC [42] (d) Deep image prior

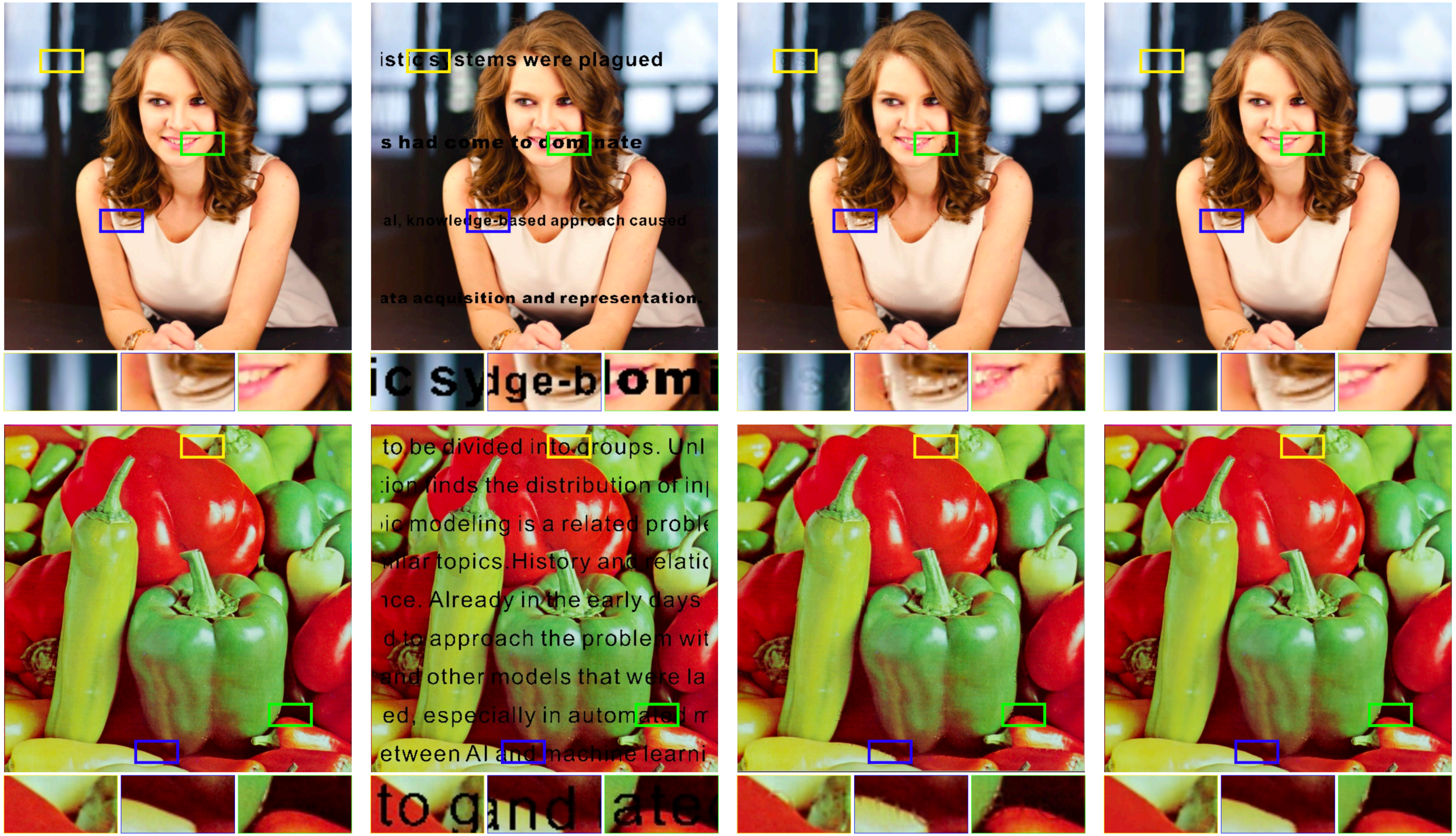
Fig. 13: Comparison with convolutional sparse coding (CSC) [42] on inpainting 50% of missing pixels. Our approach recovers a natural image with more accurate fine details than convolutional sparse coding.

Inpainting



(a) Corrupted image (b) Global-Local GAN [27] (c) Ours, LR $= 0.01$ (d) Ours, LR $= 10^{-4}$

Inpainting



(a) Original image (b) Corrupted image (c) Shepard networks [44] (d) Deep Image Prior

Выводы

- Это не новый метод решения задачи image restoration.
- Получается, что архитектура нейросети сама по себе формирует prior, который обычно получается благодаря обучению.
- Работает долго и нужна специальное отслеживание количества итераций.
- Результаты работы нейросети <u>без обучения</u> близок к нейросетям, которые были обучены на датасете.