BRNO UNIVERSITY OF TECHNOLOGY

Faculty of Electrical Engineering
and Communication

MASTER'S THESIS

Brno, 2020 Bc. Anzhelika Mezina

BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF TELECOMMUNICATIONS
ÚSTAV TELEKOMUNIKACÍ

FACE SUPERRESOLUTION FROM IMAGE SEQUENCE
SUPERROZLIŠENÍ OBLIČEJE ZE SEKVENCE SNÍMKŮ

MASTER'S THESIS
DIPLOMOVÁ PRÁCE

AUTHOR — **Bc. Anzhelika Mezina**
AUTOR PRÁCE

SUPERVISOR — **doc. Ing. Radim Burget, Ph.D.**
VEDOUCÍ PRÁCE

BRNO 2020

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Diplomová práce

magisterský navazující studijní obor **Informační bezpečnost**

Ústav telekomunikací

Studentka: Bc. Anzhelika Mezina ***ID:*** 185934

Ročník: 2 ***Akademický rok:*** 2019/20

NÁZEV TÉMATU:

Superrozlišení obličeje ze sekvence snímků

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s problematikou metod zvyšování rozlišení obrazových dat za účelem přesnější identifikace osob z video záznamů. Zaměřte se zejména na technologie neuronových sítí a zpracujte přehled aktuálních metod, které natrénujte a vzájemně srovnejte. Vytvořte vhodnou trénovací množinu obsahující sekvence snímků obličeje. Navrhněte nejméně dvě architektury neuronových sítí pracující se sekvencí snímků, navržené sítě natrénujte v prostředí Python/Tensorflow a srovnejte přesnost v porovnání s metodami zpracování jediného snímku. Naměřené hodnoty vhodně zaneste do tabulky a vzájemně srovnejte metody založené na jednom snímku ve srovnání s vícesnímkovými přístupy. Dosažené výsledky diskutujte a uveďte srovnání se stavem ve světě.

DOPORUČENÁ LITERATURA:

[1] Yu, Jiahui, et al. "Wide activation for efficient and accurate image super-resolution." arXiv preprint arXiv:1808.08718 (2018).

[2] Ledig, Christian, et al. "Photo-realistic single image super-resolution using a generative adversarial network." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

Termín zadání: 3.2.2020 ***Termín odevzdání:*** 1.6.2020

Vedoucí práce: doc. Ing. Radim Burget, Ph.D.

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

ABSTRACT

This work is focused on application of deep learning in increasing resolution of images containing face. This can be applied in different fields, including security. For example, in case of incident, the police needs to identify a culprit from the records of security camera. The aim of this work is to propose neural network models, which would work with sequence of frames, and to compare these models with existing methods for a single image super-resolution. For this purpose, a new dataset with sequences of the images with faces is created. The methods for the single super-resolution are trained on the new dataset. The new architectures for multiframe super-resolution are proposed. They are based on U-Net model. This model is successful for segmentation tasks, but it can be also applied for super-resolution tasks. To improve this architecture, the residual blocks and its modification are used. To avoid blurring effect and recover more details, the perceptual loss function is applied. In the first part of this work, the description of neural networks and overview of the architectures, which can be applied in super-resolution, is provided. The second part contains the methods for super-resolution of a single frame, multiframe, video. In the next section, there is a description of proposed architectures and description of the experiment. In the last part of the work, multiframe methods and single frame methods are compared. In the result, the proposed methods recover more details, however, some architectures produce artefacts, which can be reduced using a filter, for example, Gaussian. New methods allow to reduce the number of failed face recognition. This fact is necessary for person identification in case of incidents.

KEYWORDS

convolution network, identification, image processing, face recognition, face superresolution, multiframe superresolution, neural networks, residual learning, single superresolution, U-Net model

ABSTRAKT

Táto práce se zabývá použitím hlubokého učení neuronových sítí ke zvýšení rozlišení obrázků, které obsahují obličeje. Tato metoda najde uplatnění v různých oblastech, zejména v bezpečnosti, například, při bezpečnostním incidentu, kdy policie potřebuje identifikovat podezřelého z nahraného videa ze sledovací kamery. Cílem této práce je navrhnout minimálně dvě architektury neuronových sítí, které budou pracovat se sekvencí snímků, a porovnat je s metodami zpracování jediného snímku. Pro tento účel je také vytvořena nová trénovací množina, obsahující sekvenci snímku obličeje. Metody zpracování jednoho snímku jsou natrénované na nové množině. Dále jsou navrženy nové metody zvětšení obrázků na základě sekvence snímků. Tyto metody jsou založené na U-Net modelu, který je úspěšný v segmentaci, ale také v superrozlišení. Pro zlepšení architektury byly použity reziduální bloky a jejich modifikace, a navíc také percepční ztrátová funkce, která dovoluje vyhnout se rozmazání a získání více detailů.

První čast této práce je věnovana popisu neuronových sítí a některých architektur, jejichž modifikace mohou být použity v superrozlišení. Druhá část se poté zabývá popisem metod pro zvýšení rozlišení obrazu pomocí jednoho snímku, několika snímků a videa. Ve třetí části jsou popsány navržené metody a experimenty a v poslední části porovnaná metod založených na jednom snímku a několika snímcích. Navržené metody jsou schopny získat více detailů v obraze, ale mohou produkovat artefakty. Ty lze ale poté eliminovat pomocí filtru, například Gaussova. Nové metody méně selhávají při detekci obličejů, a to je podstatné u identifikace člověka v případě incidentu.

KLÍČOVÁ SLOVA

detekce obličeje, identifikace, konvoluční neuronová síť, multiframe superrozlišení, neuronová síť, reziduální učení, sekvence snimků, superrozlišení obličeje, U-Net model, zpracování obrazu.

MEZINA, Anzhelika. *Face superresolution from image sequence*. Brno, 2020, 79 p. Master's Thesis. Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Telecommunications. Advised by doc. Ing. Radim Burget, Ph.D.

ROZŠÍŘENÝ ABSTRAKT

Fyzická bezpečnost se stává stále aktuálnějším problémem. Použití sledovacích kamer je téměř standardem, zejména na exponovaných veřejných místech, jako jsou například obchody, banky, stavby a další. Zásadním problémem sledovacích kamer je fakt, že kvalita videa bývá zpravidla nízká, přítom v případě bezpečnostního incidentu bude policie potřebovat záznamy z kamer k identifikaci podezřelého, a to v co nejlepší kvalitě. K řešení tohoto problému se dají použít různé techniky ke zvýšení rozlišení získaného obrazu.

Klasické metody jsou založené na interpolaci, z nichž nejpopulárnější a nejjednodušší jsou bikubické anebo bilineární interpolace. V současnosti již existuje mnoho metod založených na neuronových sítích. Tyto metody můžeme rozdělit do následujících skupin: superrozlišení jednoho snímku, superrozlišení posloupnosti snímků a superrozlišení videa.

Nejčastěji se používají metody pro zvětšení jednoho snímku, protože mají vyšší účinnost. Nejznámější metody jsou Super-Resolution Convolutional Neural Network (SRCNN), Super Resolution Generative Adversarial Network (SRGAN) a Enhanced Deep Super-resolution Network (EDSR).

Existují také metody, zaměřené na superrozlišení obličeje, ale toto odvětví v oblasti počítačového vidění není příliš populární. Metody pro rekonstrukci obrazu s obličejem jsou založeny na architekturách, které se používají k superrozlišení jednoho snímku: Generative Adversarial Network (GAN), reziduální bloky, konvoluční neuronové sítě.

Konvoluční neuronové sítě jsou velice známé v oblasti počítačového vidění a našly uplatnění i v superrozlišení, příkladem toho je architektura SRCNN, která obsahuje jen tři konvoluční vrstvy, přesto dosahuje poměrně dobrých výsledků.

Další z nejpopulárnějších architektur je Generative Adversarial Network. Základ této architektury má dvě části: Generátor a Diskriminátor. Generátor je architektura, jejímž účelem je naučit se produkovat data tak, aby byla co nejpodobnější požadovanému výsledku. Diskriminátor je nejčastěji klasifikátor, který se musí naučit rozlišit reálná data od dat vygenerovaných Generátorem. GAN je v podstatě architektura založená na soutěži Diskriminátoru a Generátoru, kde se Generátor snaží oklamat Diskriminátor, zatímco Diskriminátor se snaží naučit se data rozlišit. V případě superrozlišení se docela často používá CNN nebo reziduální učení v Generátoru.

Také u superrozlišení se můžeme často setkat s reziduálním učením. Reziduální bloky umožňují přenést vstupní data skrze celý blok a přidat je do výstupu beze změn. Tato metoda umožňuje šíření gradientu a bude vhodnější používat ji v hlubokých sítích. Zmíněná architektura SRGAN je založena na GAN architektuře s použitím reziduálních bloků.

V rámci této práce byla vytvořena trénovací množina obsahující 384 sekvencí snímků obličeje. Celá množina byla rozdělena na trénovací množinu, která zahrnuje 270 sekvencí snímků, validační množinu, která obsahuje 70 sekvencí snímků a testovací množinu se 34 sekvencemi snímků. Všechny obrázky byly rozděleny do složek podle pořadového čísla v sekvenci, aby bylo jednodušší provádět trénování. Vstup do všech architektur má rozlišení 32×32 px, výstup má rozlišení 64×64 px, 128×128 px, 256×256 px pro dvojnásobné, čtyřnásobné a osminásobné zvětšení.

Dalším krokem je trénování existujících metod pracujících s jediným snímkem. Vybrány byly metody SRCNN, SRGAN, EDSR a Super Resolution Generative Adversarial Network (SRGAN). Parametry u jednotlivých modelů byly upraveny tak, aby byly schopny pracovat s novou množinou.

Dále je návrh architektur pracujících se sekvencí snímků. Základem pro všechny modely je U-Net architektura, která je primárně cílená na segmentaci obrazu, ale našla uplatnění i v superrozlišení.

V první architektuře jsou vstupní obrázky zvětšeny pomocí bikubické interpolace v rámci předzpracování. Navíc byly použity reziduální bloky, umístěné před U-Net modelem. Tyto bloky dovolují vytáhnout příznaky a použit je dál při učení sítí. Použitá ztrátová funkce je střední kvadratická chyba (MSE). Použití této ztrátové funkce způsobuje rozmazaný efekt.

Ve druhém navrženém modelu byl použit jiný přístup ke zvětšení: namísto bikubické interpolace v rámci předzpracování byly použity subkonvoluční vrstvy, s nimiž se můžeme často setkat v metodách pro superrozlišení. Dále jsou implementovány GEU bloky. GEU blok je založen na reziduálním bloku, ale místo jednoduchého přidání vstupní vrstvy do výstupu bloku je zde implementována nová vrstva, která má váhy, a tím pádem zlepšuje učení sítě. V U-Net modelech byly také implementovány modifikace: byly použity subkonvoluční vrstvy, a navíc byla ztrátová funkce MSE nahrazena percepční ztrátovou funkcí. Architektura s těmito změnami produkuje lepší výsledky. Způsobené nežádoucí artefakty můžeme eliminovat pomocí filtru.

Třetí architektura je v podstatě pokus o zlepšení druhého modelu: byly upraveny GEU bloky, U-Net model nemá subkonvoluční vrstvy jako v původním modelu, ale Upsampling vrstvy. Ve výsledných obrazcích nejsou téměř žádné artefakty.

Zatímco druhý a třetí model obsahují GEU bloky, které se nachází mezi vstupem a subkonvolučními vrstvami, čtvrtý model je má vložené mezi subkonvolučními vrstvami a U-Net modelem. Tím způsobem můžeme extrahovat více příznaků.

SRCNN má lepší výsledky podle objektivních metrik (MSE, PSNR, SSIM). Počet neúspěšných rozpoznání obličeje je u navržené architektury menší. Navíc k tomu byl proveden průzkum: skupina lidí hlasovala, který z modelů dává lepší výsledek ve vztahu podobnosti k originálu. V konečném počtu pro dvojnásobné a čtyřnásobné

zvětšení nejlépe dopadl čtvrtý navržený model. Pro osminásobné zvětšení má lepší výsledek třetí navržený model.

Podle výsledků se objektivní metriky nemusí shodovat se subjektivním vnímáním člověka. Tím pádem některé z navržených architektur mohou být použity i v praxi, v případě potřeby rekonstrukce obrázku s obličejem. Navržené architektury jsou schopné lépe rekonstruovat detaily v obrazu a mají navíc méně nepovedených detekcí obličeje, což zvyšuje úspěšnost u praktického použití v rámci vyšetřování incidentu.

DECLARATION

I declare that I have written the Master's Thesis titled "Face superresolution from image sequence" independently, under the guidance of the advisor and using exclusively the technical references and other sources of information cited in the thesis and listed in the comprehensive bibliography at the end of the thesis.

As the author I furthermore declare that, with respect to the creation of this Master's Thesis, I have not infringed any copyright or violated anyone's personal and/or ownership rights. In this context, I am fully aware of the consequences of breaking Regulation § 11 of the Copyright Act No. 121/2000 Coll. of the Czech Republic, as amended, and of any breach of rights related to intellectual property or introduced within amendments to relevant Acts such as the Intellectual Property Act or the Criminal Code, Act No. 40/2009 Coll., Section 2, Head VI, Part 4.

Brno ....

author's signature

ACKNOWLEDGEMENT

I would like to thank my supervisor doc. Ing. Radim Burget, Ph.D. for his great leadership, time at consultations, patience, motivation to carry out the work and suggestions how to improve it.

Tato práce vznikla jako součást klíčové aktivity KA6 - Individuální výuka a zapojení studentů bakalářských a magisterských studijních programů do výzkumu v rámci projektu OP VVV Vytvoření double-degree doktorského studijního programu Elektronika a informační technologie a vytvoření doktorského studijního programu Informační bezpečnost, reg. č. CZ.02.2.69/0.0/0.0/16_018/0002575.

EVROPSKÁ UNIE
Evropské strukturální a investiční fondy
Operační program Výzkum, vývoj a vzdělávání

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

Projekt je spolufinancován Evropskou unií.

Contents

List of Figures

List of Tables

Introduction

Recent years, the physical security is one of the actual problems. The application of security cameras is getting more popular. We can meet with security cameras in a lot of places: in banks, offices and on the buildings, where they monitor the environment. The main problem is a low quality of records and in case the incidents happens, it is necessary to identify the culprits. Nowadays it's getting possible to increase the resolution of images and to improve the quality of them. Moreover, it is possible to identify the person from a video record.

Face super-resolution is one of the challenges in the computer vision. Since face detection and recognition are actively used in the new technologies, it is important to have available methods, which we can use to reconstruct the image of the face without distortion. This technique can be applied in security field. One of the applications is a reconstruction of the face from a video. In case of incidents it can be useful for the police. Especially, when it is not possible for a human to recognize the face of the culprit in video record. Another application is biometric identification. Lately, the technology of face identification is getting more popular and it is used, for example, in smartphones to unblock a device or in systems for access control in buildings and for monitoring streets to identify a suspect [1].

This work focuses on increasing image resolution. The traditional way is to use the nonlinear bicubic interpolation. The advantage is a computational simplicity, which makes it relatively fast and allows to process real-time images. On the other hand, these methods produce aliasing and blurring [2]. There are a lot of approaches, which we can use to increase image resolution using neural networks. All approaches can be summed up into three main categories: single image super-resolution, multi-frame super-resolution and video super-resolution.

The main contribution of this thesis is utilizing the U-Net architecture for image reconstruction with a face from a sequence composed of 6 frames. The proposed architectures are based on the U-Net model. Moreover, these architectures are tested for $\times 2$, $\times 4$ and $\times 8$ upscaling. According to the results, these methods allow to reconstruct more details for 8 scale factor than the methods for single frame super-resolution and allow us to recognize the face of person.

In the first chapter the neural networks are defined and chosen well-known architectures for image processing are described. In the second chapter the existing approaches and relevant methods for single-frame super-resolution, multi-frame super-resolution and video super-resolution are introduced. In the third chapter the steps of experiment with specification of created dataset and implemented models for multi-frame super-resolution are described. In the last chapter the results of the experiment are shown and discussed.

1 Neural network models

Nowadays, deep learning and neural networks take a big place in computer science. They give solutions for different problems, for example, in computer vision, medicine, security, speech recognition, natural language processing. Moreover, neural networks are actively used in real life: a face recognition for security systems, a voice assistants in smartphones, recommendation systems on websites. The neural networks are really useful for people, since they can make the life more comfortable and perform tasks faster than humans.

1.1 Definition

Artificial neural networks were inspired by the architecture of animal brains. The network receives the input signals and produces output signals. It consists of simple elements such as neuron, which are connected and create the system of neurons – network. The definition for artificial neural network was formulated by Dr. Robert Hecht-Nielson: "...a computing system made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external inputs" [3].

1.2 Neuron

The artificial neuron has a couple of inputs, weights, bias and activation function. The scheme of the neuron is shown in Figure 1.1. The neuron receives the information, computes the sum of bias b, which is usually a constant, and set of input values x_n, which were multiplied with associated weights ω_n on the connection. The result is passed through the activation function f to produce the output y [3].

The activation function is used to determine, whether the neuron should be activated or not. The most often used activation functions are sigmoid, hyperbolic tangent (tanH), Rectified Linear Unit (ReLU), Leaky ReLU and softmax.

Sigmoid. Sigmoid activation function is a non-linear function, which is shown in Figure 1.2. This function is very useful for the prediction of probability, since its ranges are between 0 and 1. However, for large negative input values the function returns 0 and for large positive input values it returns 1. It can cause a vanishing gradient problem, because with large positive or negative x, the difference between y is really small. This can lead into small gradient changes, consequently, to slow training of the network or not training it at all [4]. The mathematical formula of

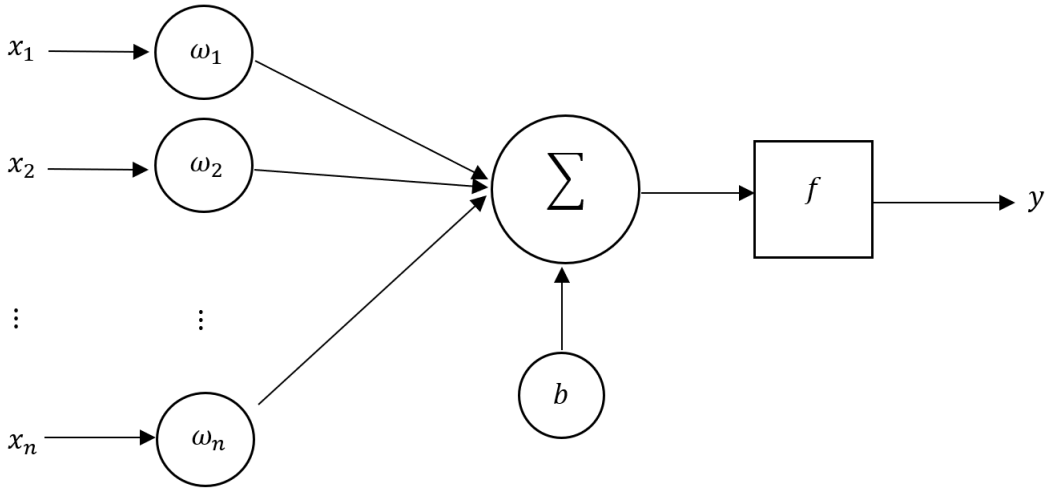


Fig. 1.1: Artificial neuron, where x_n – input values, ω_n – weights, b – bias, f – activation function, y – output.

sigmoid activation function is depicted below [5].

$$f(x) = \frac{1}{1 + \mathrm{e}^{-x}} \tag{1.1}$$

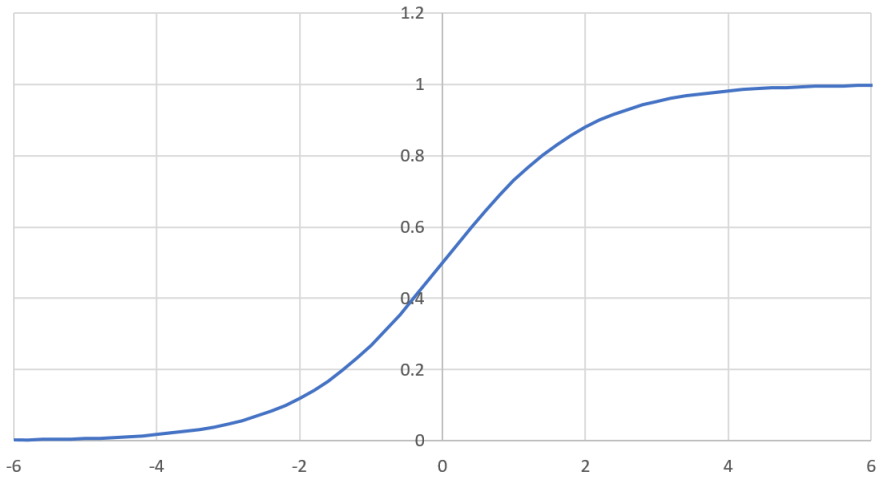


Fig. 1.2: Sigmoid function.

Tanh. As it can can be seen in the Figure 1.3, this function looks like a sigmoid function. The main difference here is the range of output, which is between -1 and 1. Output is zero-centered, consequently, it is possible to get the output with a different sign and determine, which neuron should be considered and which will be ignored. However, the tanh function still has the vanishing gradient problem. The

function has a mathematical formula, which is shown below [5].

$$f(x) = \frac{2}{1 + e^{-2x}} - 1 \tag{1.2}$$

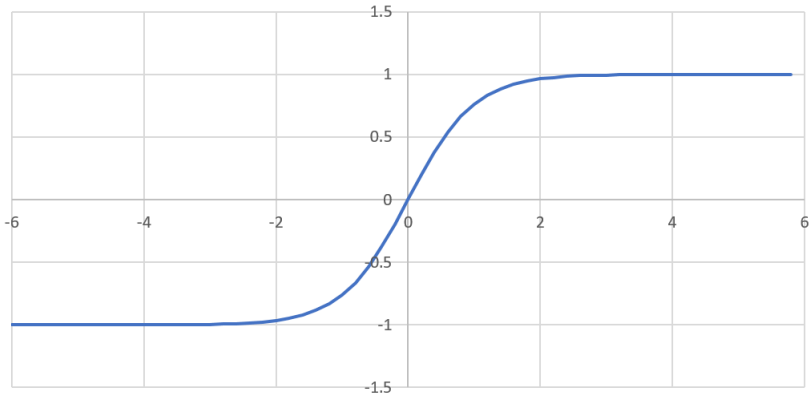


Fig. 1.3: Tanh function.

ReLU. One of the most frequently used activation functions is ReLU. It will return the zero, if the input value is negative, and the same value as input, if the input value is positive. The plot is shown in the Figure 1.4. Because of its non-linearity, it also doesn't have the problem with back-propagation. Moreover, it solves the problem of the vanishing gradient, because of positive values, the gradient has a constant value. On the other side, this function has a problem of "Dead ReLU": the neurons with negative input values to the activation function become inactive. The mathematical formula of this function is [5]:

$$f(x) = max(0, x) \tag{1.3}$$

Leaky ReLU. The modification of ReLU is Leaky ReLU, which solves the problem of "Dead ReLU". Leaky ReLU, which is shown in Figure 1.5, allows a small non-zero gradient, when the neuron is not active [6]. The range of the ReLU function is not limited. The value of function for negative inputs is multiplied with a small factor, usually it is 0.01. The mathematical formula of the LeakyReLU is [5]:

$$f(x) = max(0.01 \cdot x, x) \tag{1.4}$$

Softmax. This function is supposed to be used for the multi-class classification. The aim of the softmax function is to normalize the output values to the range between 0 and 1. It returns the vector with the probabilities of each class and the sum of all of them is 1. The softmax function can be written as follows [5]

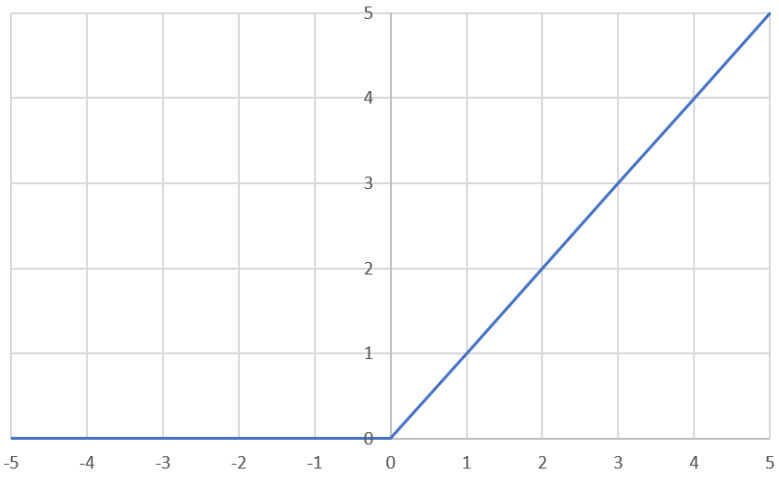


Fig. 1.4: ReLU function.

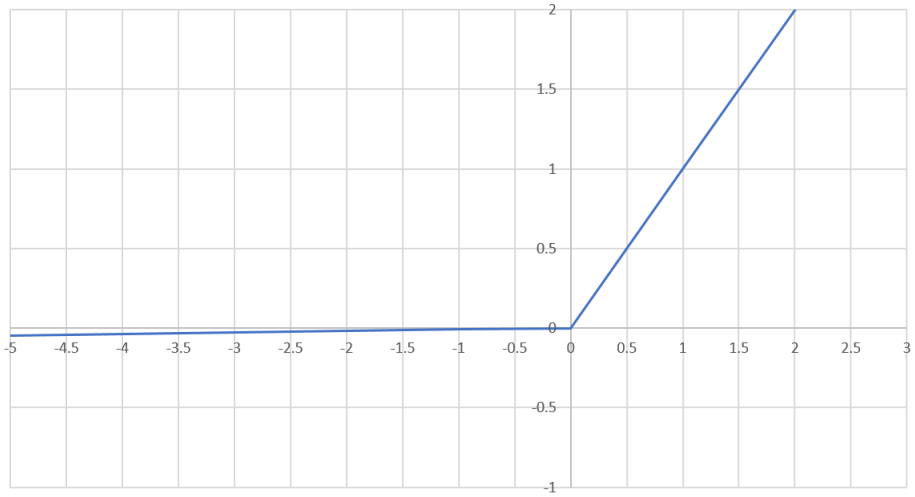


Fig. 1.5: Leaky ReLU function.

$$\sigma(z)_j = \frac{\mathrm{e}^{z_j}}{\sum_{k=1}^{K} \mathrm{e}^{z_k}}, \tag{1.5}$$

where

- z – vector of the inputs,
- j – index of output units,
- K – number of the output values.

1.3 Layers

The groups of neurons create layers. The neurons from one layer are connected with neurons from another layer, and those connections create the architecture of neural network.

Input layer

Input layer is the first layer in the neural network and contains the initial data. There's no operation in this layer, it only receives the input signal and passes it on to the next layer in the network.

Convolutional layer

The aim of this layer is to extract features from the previous layer. This layer makes the convolution operation on the input data.

The convolutional layer has kernel – an array of weights, size of it is smaller than an input image, for example 3×3, 5×5. The larger kernel size, for example 7×7, is used for extraction of more significant features of the input image. However, more memory is required to process this layer. This kernel has a stride parameter, which defines the step for kernel, usually it is 1. During convolution, the kernel slides step by step on the input values. The currently used part of the input is called receptive field. Each value of the receptive field is multiplied with a corresponding value from the kernel. The sum of all multiplies is written as a new pixel value. The scheme of this operation is shown in Figure 1.6 [7].

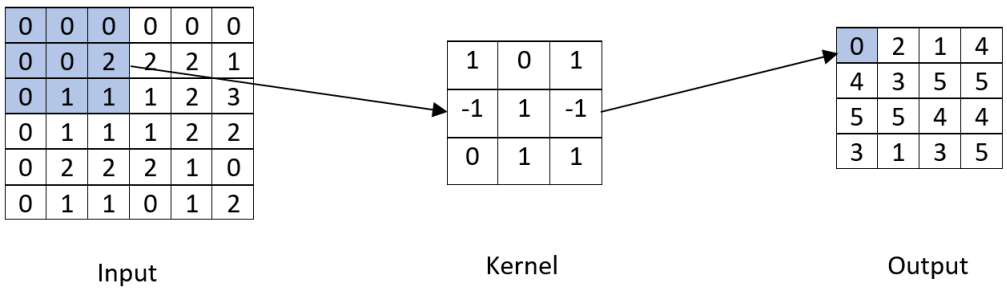


Fig. 1.6: Convolution.

Max pooling layer

The aim of this layer is to reduce the spatial size of a input tensor. For example, the filter has the size 2×2 and a stride is 2. It means, that the parts of the input

have 2×2 and the "step" of sliding is 2. It works as follows: each part of input is reduced to a single pixel by choosing the maximum value (Max function). The depth dimension remains unchanged. The main advantage of the operation is a reduction of needed used memory, that allows to train the network faster [8].

Batch normalization layer

The distribution of all inputs of a layer are changed during the training. That's the reason to choose lower learning rate and carefully pick the parameter of the initialization in order to prevent "internal covariate shift". The aim of this layer is to be a part of the architecture, which performs normalization for each training mini-batch. This layer allows to use a higher learning rate. Moreover, the batch normalization can be a regularizer and eliminates the need for dropout [9].

Fully connected layer

This layer is usually the last one in the architecture. The aim of a fully connected layer is to use the extracted features from previous layers to classify the image into a label. For example, if the features contain such things, like the wheels, the headlights, the high probability should be assigned to the label “car”. The output of this layer is a vector with values of the probabilities for each class, which the image belongs to. The used activation function for this layer is softmax, which normalizes the values to the range from 0 to 1. This layer is usually used in image classification [10].

1.4 Convolutional Neural Network

One of the most popular architecture for image processing is Convolutional Neural Network (CNN). It is used for image recognition, classification, super-resolution, object detection and others.

CNNs take an inspiration from the visual cortex. Visual cortex has small areas with the cells, which are responsible to process specific regions of the visual fields. For example, some of them are fired for vertical and horizontal edges. Visual perception is produced with neurons, which are arranged into columns. All of these principles are used in CNNs.

For human it is not a problem to identify the surrounding environment, because human brain learns the things since birth naturally. However, it is difficult to share this skill with machines: computer just "sees" the input image which is represented as an array of pixel values. Conventional tasks for a computer can be image classification, object detection or denoising images.

As it was mentioned before, the convolutional networks are inspired by human brain and used for image processing tasks. The structure of CNN is multilayer.

The backpropagation or backward propagation is often used to calculate the error contribution of each neuron after a batch of data is processed. The main idea is to calculate the error at the output and to distribute it back through the network. It is usually used by gradient descent to adjust the weight values of neurons during calculation of the loss function. The correct known output is required for this method, that is why it is used for supervised learning [11].

The most architectures have the combination of these layers: convolutional layers, max pooling layers, activation (ReLU) layers and fully connected layer, which are described in Section 1.3. The typical CNN has this structure: Input -> [[Conv -> ReLU]*N -> Pool (optionally)]*M -> [Fully connected -> ReLU]*K -> Fully Connected, where * – operation of repetition, N, M, K – number of repetition [8]. The illustration of architecture is shown in Figure 1.7.

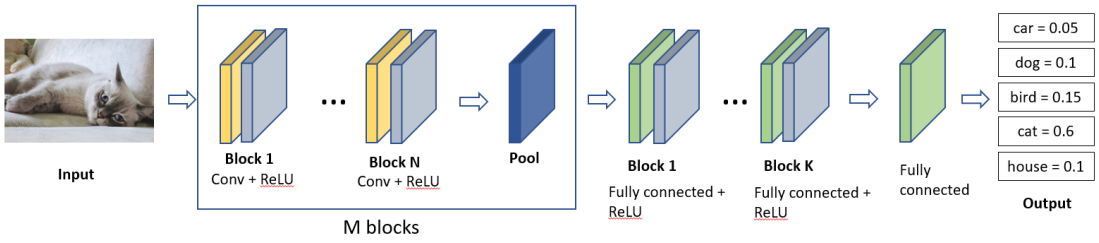


Fig. 1.7: CNN architecture.

The main advantage of this kind of network is an utilization of the convolution operation. It allows us to extract features from specific to abstract level. Moreover, the architecture can produce the hierarchy of features, filter the unnecessary details and keep the important ones.

The disadvantage of CNN is that it usually contains a lot of layers, which makes it a deep network. This fact makes difficulties to understand how the features are extracted and to debug the network in case of error. However, if the produced results don't satisfy the expectations, the problem can be in incorrect parameters or too small dataset.

1.5 U-Net

The U-Net architecture was introduced in 2015 for biomedical image segmentation. The researchers actively utilized and modified the original architecture for different

tasks, including the image super-resolution [12]. The U-Net architecture is shown in Figure 1.8.

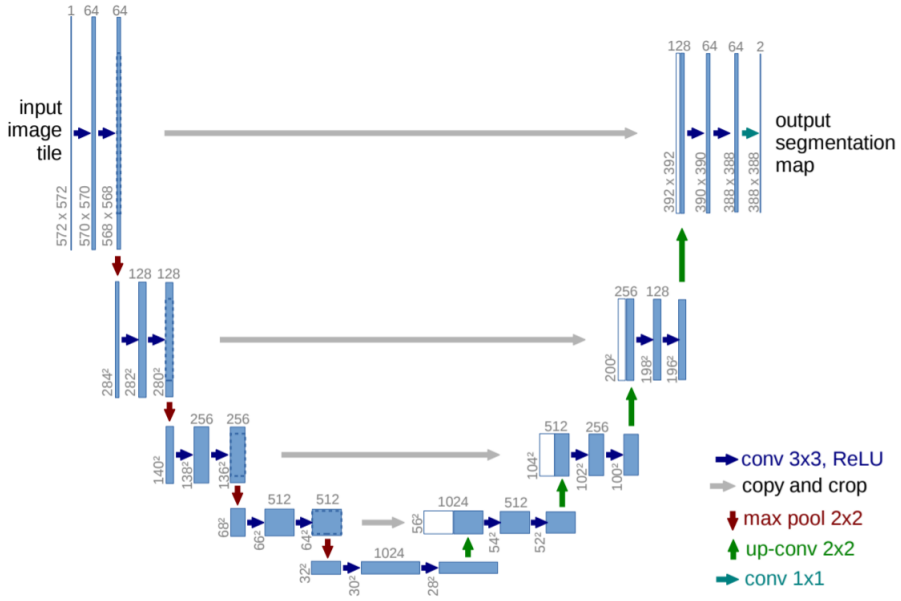


Fig. 1.8: U-Net architecture. Source [13].

The proposed architecture is based on Fully Convolutional Network (FCN). However, this architecture was modified and extended to work with very few training images. The main idea in FCN is to extend a usual contracting network by successive layers using pooling operators instead of upsampling operators. It allows to increase the resolution of the output.

U-Net has two parts: a contracting path (left side) and an expansive path (right side). The contracting part is a typical convolutional network. It contains the blocks with the same layout. Each block has two convolutional layers with kernel 3×3, each followed by a ReLU, and 2×2 max pooling layer with stride 2 for downsampling. The number of feature maps doubles after each block. The expansive path has repeated blocks, which have upsampling layer, convolutional layer, a concatenation with the feature maps from the contracting path, two convolutional layers with kernel 3×3, each followed by ReLU activation function.

For mapping to the desired number of classes the last convolutional layer with kernel 1×1 is used [13]. Moreover, the important modification is a large number of feature channels in the upsampling part.

However, the U-Net model is used for segmentation not only of biomedical images, but also of images from satellites [14]. This architecture was also combined with residual blocks and used for super-resolution task [12].

1.6 Deep Residual Learning

In recent years, residual blocks are often used for computer vision tasks, including super-resolution. The examples of application can be found in these articles: [15], [16], [17], [12]. The mentioned residual block is shown in Figure 1.9.

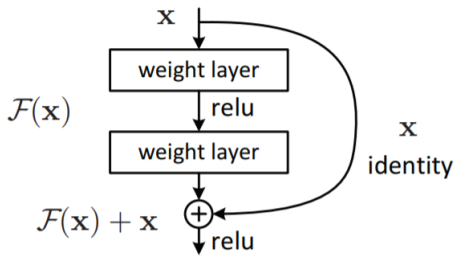


Fig. 1.9: Residual block. Source [18].

It is realized with "shortcut connections", which are skipping one or more layers. In case of demonstrated one in Figure 1.9, the outputs of the shortcut connections are added to the outputs of the stacked layers. The shortcut connections don't have any extra parameters and they don't add any computational complexity. The deep residual learning can be applied when deeper networks start converging and accuracy gets saturated and then degrades rapidly [18]. The main difference from traditional networks is getting information not only from the previous layer, but also use the information from the steps behind layer. This method helps to train deep networks with increasing accuracy. Moreover, this block is flexible and there can be used not just two layers, but also three or more.

The proposed network with residual blocks is called ResNet, which was introduced in 2015 by Microsoft researchers. The aim of this network is to help to train the deep neural network. The performance gets saturated or even starts degrading rapidly because of vanishing gradient problem.

The proposed architecture was inspired by VGG network[1] [19] and has 34 weighted layers. This network applies the following rules: firstly, if the output feature map size is the same as in previous layer, the layers have the same number of filters; and secondly, the number of filters is doubled, if the feature map size is halved. At the end of the network the 1000-way fully-connected layer with softmax is used. The authors added the shortcut connections, which make the network residual. These connections can be used if the dimensions of input and output are the same. In case of different dimensions, the extra zero entries padding can be used for increasing

[1] it is developed by Visual Geometry Group

dimensions or the projection shortcut is used to match dimensions. The proposed architecture is shown in Figure 1.10.

After a number of experiments, the authors got to conclusion, that with increasing number of layers, the accuracy is increasing too. The model with 152 layers outperformed the previously existing methods [18].

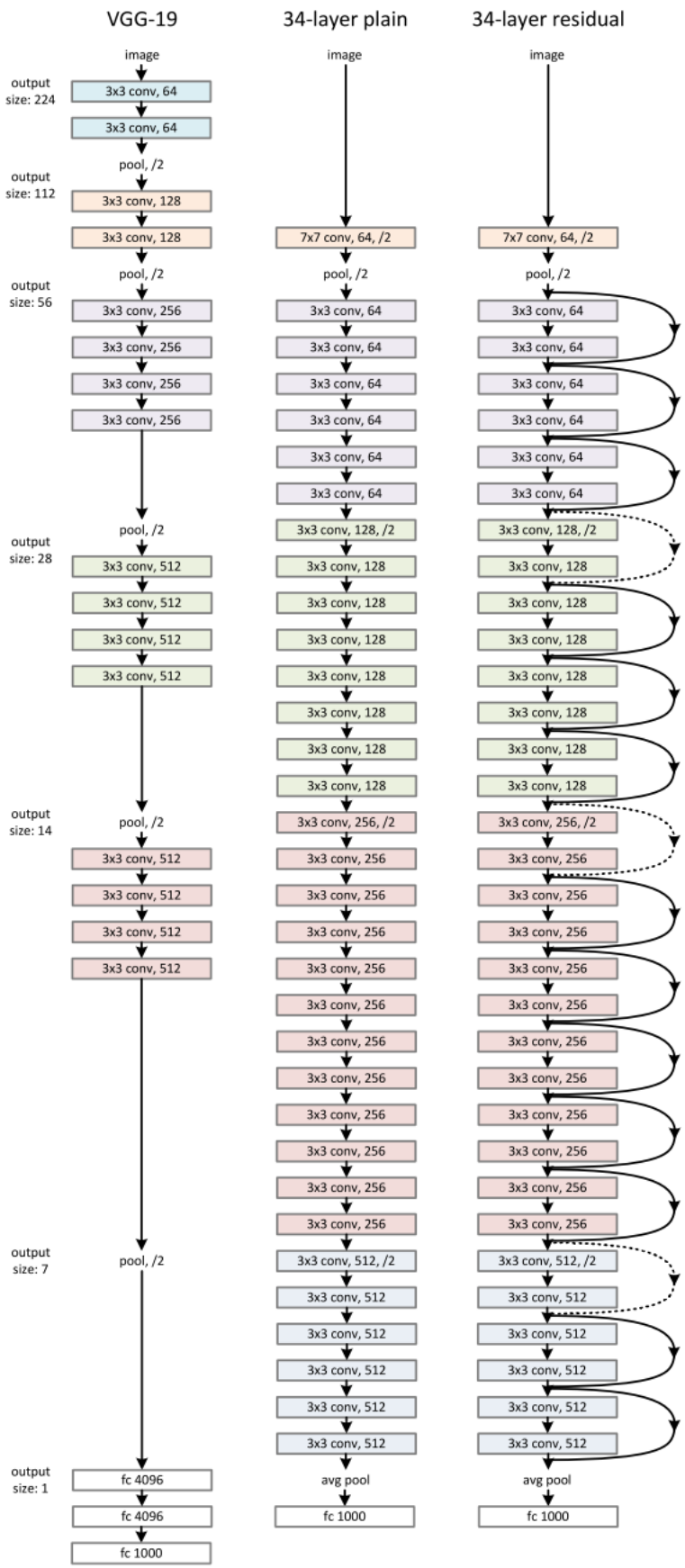


Fig. 1.10: Residual network. Source [18].

1.7 Generative Adversarial Network

Generative Adversarial network (GAN) is another family of neural networks, which are very popular recent years. The main difference from the traditional networks is that not only one network is trained, but two networks are trained simultaneously. These two networks are known as generator and discriminator.

The aim of the generator is to be able to process input noise and to reproduce the output similar to realistic data. However, the goal of discriminator is to learn to determine, whether input data is real or generated. The discriminator behaves as classifier and it's the output is a probability of the data being real. In this case, the discriminator is trained to maximize the probability $D(x)$ of assigning the correct label. However, the generator is trained to minimize $\log(1 - D(G(z)))$, where $G(z)$ is output of the generator, z is the input noise to the generator. The algorithm of GAN is shown in Figure 1.11.

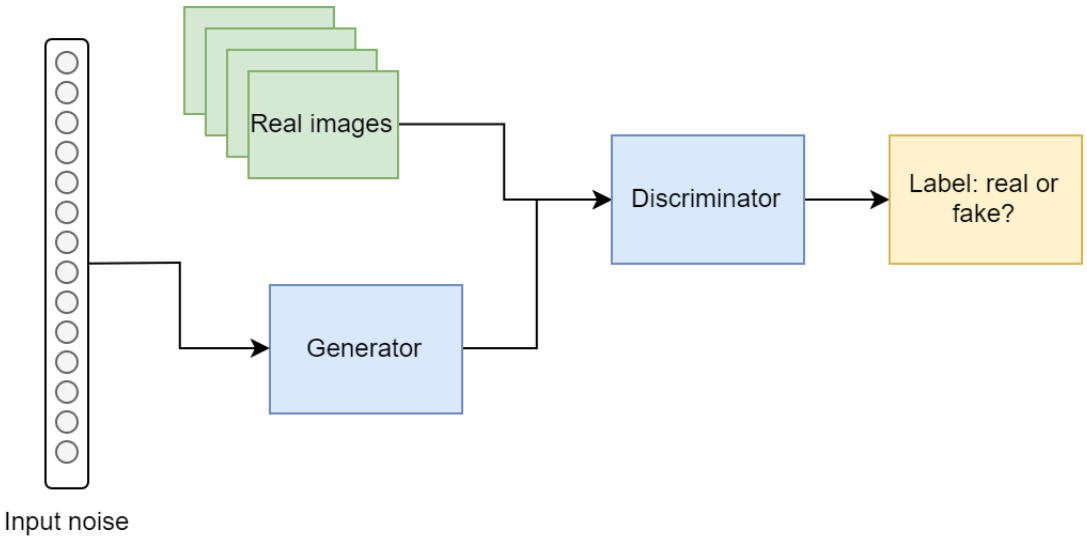


Fig. 1.11: GAN architecture.

The training of GAN can be characterized as two-player minimax game with value function $V(G, D)$:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))], \qquad (1.6)$$

where

- $p_{data}(x)$ – generator's distribution over data x,
- $p_z(z)$ – input noise variables,
- $D(x)$ – the probability that x came from the data rather than the generator,
- G – a differentiable function represented by a multilayer perceptron,

- D – multilayer perceptron that outputs a single scalar.

During training, the parameters of one model are updated, while the parameters of the other are fixed. Ideally, the discriminator is trained until the optimal state is reached and then the generator is updated. However, it is computationally prohibitive and in some cases can lead to overfitting. To avoid this, the discriminator is trained for a small number of iterations and after that the generator is updated [20].

There can appear some problems, for example, vanishing gradients. If the discriminator is trained very good, the generator will not get enough information to be trained more accurately [21]. Another problem is collapsing mode. It means, that generator learns to produce only one output, instead of generating different ones. The discriminator processes each input independently, and if the collapse has occurred, the discriminator learns, that this input is from generator and always rejects it. The possible solution to this problem is using multiple input to discriminator, that allows discriminator to take a look at the data in combination [22].

The recent years, there were a lot of approaches based on GAN, including face super resolution [23], anomaly detection in network [24], image-to-image translation [25], face swapping [26].

2 Super-resolution

2.1 Definition

Super-resolution (SR) is a process of upscaling and improving quality of images. The traditional methods are based on generating a high resolution image from a low resolution (LR) image using the different techniques [27]. The main challenge is to recover the missing information in a low resolution image. It's considered, that the low resolution image is downsampled with the bicubic method. However, other degradation factors, such as blur and noise, also should be considered for a practical usage [28]. There has been a lot of approaches since last years such as frequency domain approaches, interpolation-based approaches, machine learning approaches. The latter shows the best results among the others. The often used architectures are Convolutional Neural Network (CNN) [29], Generative Adversarial Networks (GAN) [15], Residual Dense Network (RDN) [30]. Moreover, the image super-resolution can be applied to a different number of input images: using only one frame (single super-resolution) and using a couple of frames to get the one reconstructed image (multi-frame super-resolution). The single-frame image super-resolution (SISR) has a high efficiency, that's why it's more popular than multi-frame image super-resolution (MISR). On the other hand, MISR allows to get more information from the images of the same scene, and it can give the final image more accurately. This method is mainly used for increasing a quality of video. The video super-resolution is also field of the interest for researchers, especially to process video in real-time.

2.2 Single-frame Super-resolution

Single image super-resolution is challenging and ill-posed problem, because the low resolution image has lost high frequency information, which is responsible for sharpness and details. Moreover, large enlargement task in SR, such as $\times 8$ enlargement, is especially challenging due to much more high frequency information being lost comparing to other small enlargement tasks, for example $\times 2$, $\times 4$ enlargements. There are some approaches for single image SR, which are based on deep learning, such as Super-Resolution Convolutional Neural Network, Very Deep Super-resolution, SRResNet, Enhanced Deep Super-Resolution network and Dense Deep Back-Projection Network, which are described in Subsection 2.2.1. Dense Deep Back-Projection Network is a state-of-the-art for scaling factor $\times 8$ [31].

One of the branches of the single-frame super-resolutions is a face super-resolution. This is also a challenging task, since it requires more attention to details to be recovered. However, there's not so much approaches comparing to the standard

single-frame super-resolution. Moreover, face super-resolution approaches are based on encoder-decoder architectures and GANs. That is why, in this part of the work, the main focus is on single-frame super-resolution, since it is a basis for the most super-resolution tasks.

2.2.1 Architectures

Super-Resolution Convolutional Neural Network

Super-Resolution Convolutional Neural Network (SRCNN) is the deep learning method for image super-resolution, which has been introduced in 2014 [29]. Convolutional Neural Network is used for computer vision tasks, but the SRCNN is modification of it and it is primary used for image super-resolution. This architecture has only three parts: patch extraction and representation, non-linear mapping and reconstruction. Before image will pass through the network, it needs to be upscaled with bicubic interpolation. The architecture of this model is demonstrated in Figure 2.1.

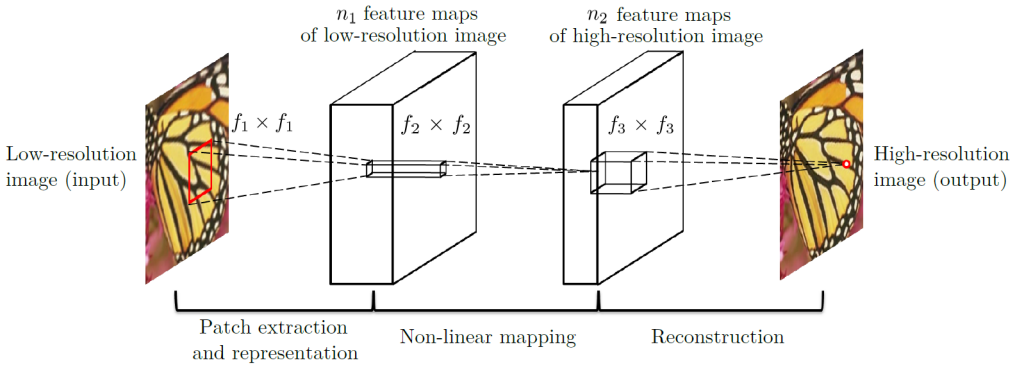


Fig. 2.1: SRCNN architecture. Source [29].

Patch extraction and representation. This operation extracts features from the low resolution image, which were upscaled to desired size. The first step is formulated as

$$F_1(Y) = max(0, W_1 * Y + B_1), \tag{2.1}$$

where

- Y – upscaled LR image,
- W_1 – n_1 filters of size $c \times f_1 \times f_1$,
- c – number of channels,
- f_1 – the spatial size,
- B_1 – n_1-dimensional, whose each element is associated with a filter,

- $*$ – convolution operation.

Non-linear Mapping. In this part, the mapping of the extracted features from low resolution images to high resolution image patches is provided. This step is described as

$$F_2(Y) = max(0, W_2 * F_1(Y) + B_2), \tag{2.2}$$

where

- W_2 – n_2 filters of size $n_1 \times f_2 \times f_2$,
- B_1 – n_2-dimensional biases.

Reconstruction. This operation generates the final HR image from extracted patches from previous layer. This step can be formulated as

$$F(Y) = W_3 * F_2(Y) + B_3, \tag{2.3}$$

where

- W_3 – c filters of size $n_2 \times f_3 \times f_3$,
- B_3 – c-dimensional biases.

The used loss function is Mean Square Error (MSE), which is computed as

$$MSE = \frac{1}{n} \sum_{i=1}^{n} ||F(Y_i; \theta) - X_i||^2, \tag{2.4}$$

where

- $F(Y_i; \theta)$ – the reconstructed images,
- X_i – a set of high-resolution images,
- Y_i – a set of corresponding low-resolution images,
- θ – estimated parameters $W_1, W_2, W_3, B_1, B_2, B_3$,
- n – the number of training samples.

Super-resolution Generative Adversarial Network

Super-resolution Generative Adversarial Network (SRGAN) is based on GAN, which is described in Section 1.7 and uses a deep residual network (ResNet), which is described in Section 1.6. Moreover, a novel perceptual loss, which uses high-level feature maps of the VGG network, is defined. The main aim is to train generator to produce the high resolution image from input low resolution one. The general idea of GAN is to train generator to fool the discriminator, which is trained to differ generated super-resolution images from the real images. That's the reason for generator to learn to generate images, which are very similar to the real ones. There is the SRGAN architecture in the Figure 2.2.

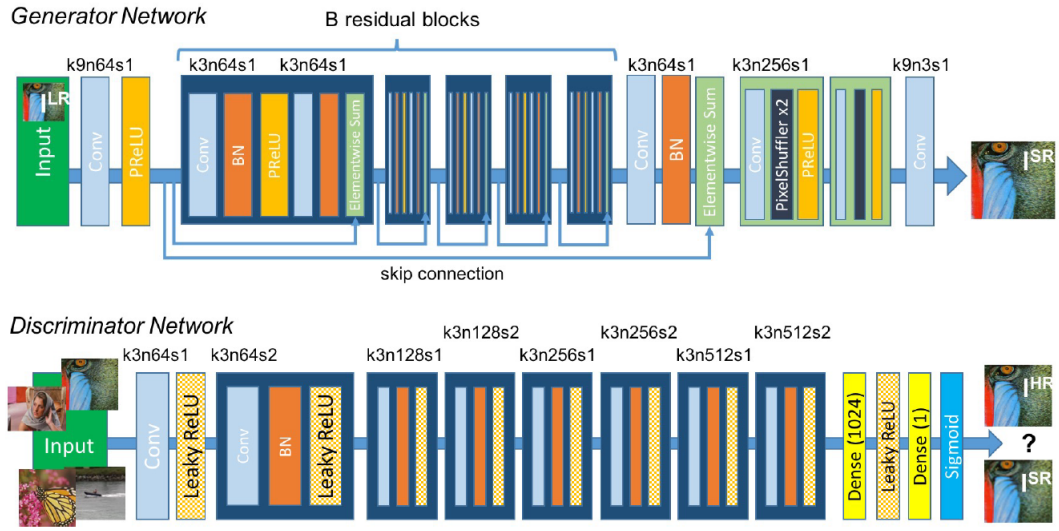


Fig. 2.2: SRGAN architecture, where k – kernel size, n – number of feature maps, s – stride. Source [15].

The generator consists of residual blocks. Each block has the same layout, which contains two convolutional layers, batch normalization layers and activation layer with ParametricReLU function. The upsampling of image is done with two sub-pixel convolutional layers.

A goal of the discriminator is to learn to discriminate between real HR images and generated SR images. It contains eight convolutional layers with an increasing number of filter kernels, increasing from 64 to 512 as in the VGG network [15]. Two dense layers are used after convolutional layers and a final sigmoid activation function is used to obtain a probability for classification if there's HR or generated SR image.

Another important thing is a perceptual loss function. It is defined as a weighted sum of content loss and adversarial loss, and it can be defined as follow:

$$l^{SR} = \underbrace{\underbrace{l_X^{SR}}_{\text{content loss}} + \underbrace{10^{-3} l_{Gen}^{SR}}_{\text{adversarial loss}}}_{\text{perceptual loss}} . \tag{2.5}$$

Content loss. Since MSE loss function has the problem with lacking high frequency content, which provides more details in images and makes it sharper, there was defined the VGG loss, which is based on pre-trained 19-layers VGG network. This loss is defined as:

$$l^{SR}_{VGG/i.j} = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} (\phi_{i,j}(I^{HR})_{x,y} - \phi_{i,j}(G_{\theta_G}(I^{LR}))_{x,y})^2, \tag{2.6}$$

where

- $\phi_{i,j}$ – the feature maps obtained by j-th convolution before i-th maxpooling layer within VGG19 network,
- I^{LR} – low resolution image,
- I^{HR} – high resolution image,
- $W_{i,j}$, $H_{i,j}$ – dimensions of the feature maps.

Adversarial loss. This part of perceptual loss is a generative component, which encourages the network to give preference to natural images in order to fool the discriminator network. The adversarial loss is defined as follows:

$$l^{SR}_{Gen} = \sum_{n=1}^{N} -\log D_{\theta_D}(G_{\theta_G}(I^{LR})), \tag{2.7}$$

where

- $D_{\theta_D}(G_{\theta_G}(I^{LR}))$ – probability that reconstructed image is a natural high resolution image,
- $G_{\theta_G}(I^{LR})$ – reconstructed image by generator.

The main focus of work in [15] was to show the perceptual quality of super-resolved images rather than computational efficiency. Also, it was shown, that standard quantitative measures (MSE, PSNR) don't catch all details with respect to human visual system. The subjective metrics, Mean Opinion Score (MOS), shown, that SRGAN makes better reconstruction, than other methods, for example: SRResNet, SRCNN, Deeply-Recursive Convolutional Network (DRCN). However, this model is not optimized for real-time video SR [15].

Enhanced Deep Super-resolution Network

As it was mentioned in the section describing the previous architecture, the SRResNet uses the deep residual network almost without any changes. And even it shows a good performance results, the architecture still can be optimized for the SR problem, since originally ResNet had the goal to solve the computer vision tasks such as object detection, classification, segmentation [18]. As the result of the optimization and some modifications of SRResNet, a new architecture, called Enhanced Deep Super-Resolution Network (EDSR), was proposed.

First of all, residual network from the SRResNet was analysed and the unnecessary modules were removed to simplify the network. For example, in ResNet and SRResNet the residual block was used with the following structure: convolutional layer – batch normalization – ReLU activation – convolutional layer – batch

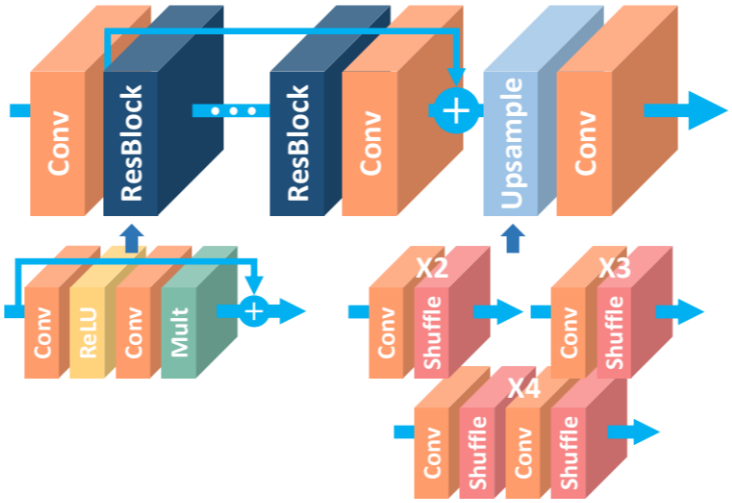


Fig. 2.3: EDSR architecture. Source [16].

normalization. But the batch normalization layers were removed in this proposed version and the used structure is: convolutional layer – ReLU activation – convolutional layer. The modified version outperforms the original one and also decreases the GPU memory usage [16]. The architecture is shown in Figure 2.3.

Secondly, a lot of approaches solve the different scale factors as independent problems, but don't utilize mutual relationships among different scales in super resolution. However, the proposed model trains the high scale model from the pretrained low scale models and shares parameters across different scales. In this way, the training for $\times 3$ and $\times 4$ scaling uses the pretrained model for $\times 2$. This approach accelerates the training and improves the final performance.

Edge Enhanced Single Image Super-resolution

Since existing methods usually minimize a loss between the output SR image and the ground truth image, they yield very high peak signal-to-noise ratio (PSNR). Unfortunately, minimization of this loss leads to blurred edges due to averaging of possible solutions. That's the reason to focus on an improvement of edge detection. SREdgeNet consists of 3 sequential deep neural network modules.

The first module upscales input image. For this purpose, any state-of-the-art SR network can be used. EDSR is selected in the proposed architecture. The original EDSR was modified by incorporating pyramid polling into the SR network. Firstly, average pooling is performed and convolutions for each of the four pyramid scales are executed. 6×6 pyramid pulling are used to receive more enlarged information.

The second module is any edge detection network. In this approach the DenseEdgeNet is proposed, which is shown in Figure 2.4. The DenseEdgeNet is trained with high resolution images, because it's necessary to get the correct information about the edges. After that, the image from first step goes through this trained network. DenseEdgeNet uses Canny edge detector.

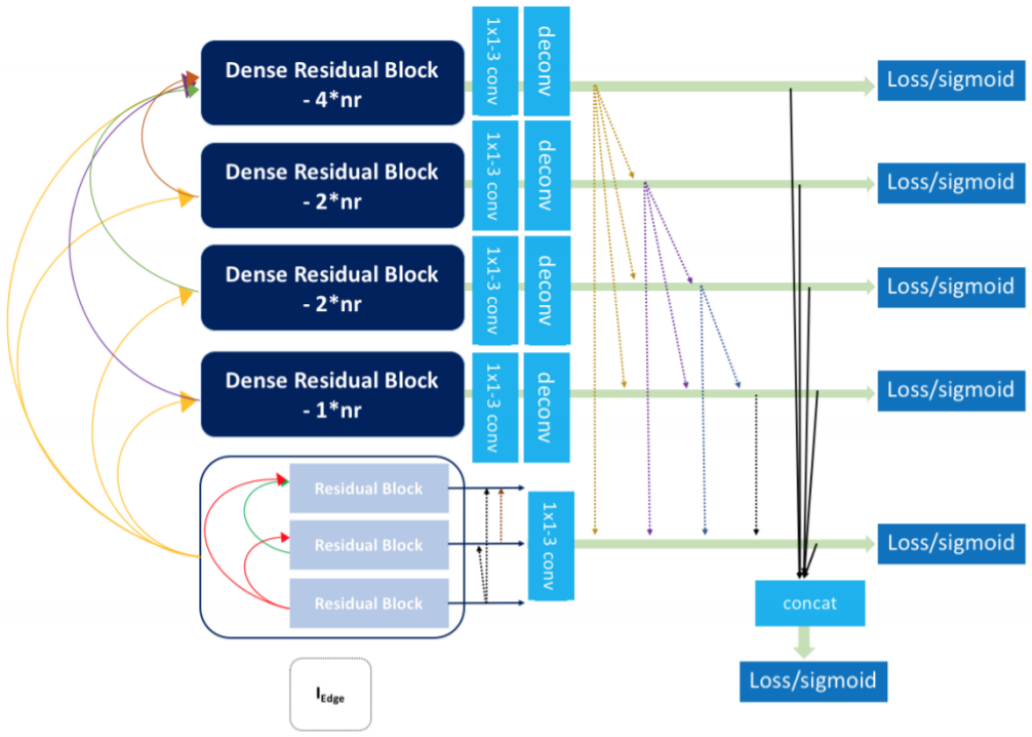


Fig. 2.4: DenseEdgeNet architecture for edge detection. Source [31].

In the third module the outputs of the first and second modules are merged with a proposed MergeNet, which is based on EDSR with the reduced number of parameters by half [31]. The architecture of MergeNet is shown in Figure 2.5. The main differences from the original EDSR are input of 4 channels (RGB image with Edge) and edge skip connection, which refers to connecting the edges of an input to residual learning. This method helps to efficiently use the edge information. The MergeNet uses 128 features maps instead of 256 and 16 residual blocks instead of 32.

Comparing with SRGAN, the model recovers more sharp edges and overall patterns. The quantitative results in terms of PSNR or SSIM are also better than results of SRGAN. Qualitatively, other networks generate noticeable artifacts and blurred edges. However, SREdgeNet network can recover sharper and clearer edges. According to results, it outperformed previously mentioned state-of-art methods.

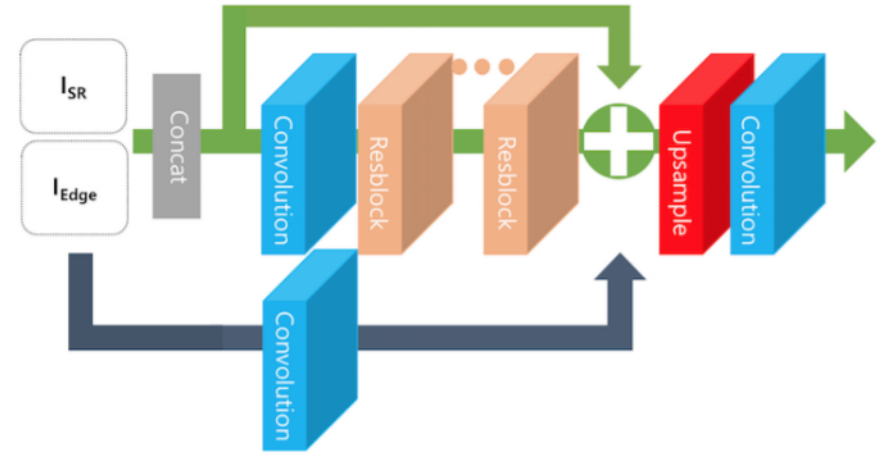


Fig. 2.5: MergeNet architecture. Source [31].

Enhanced Super-Resolution Generative Adversarial Networks

SRGAN gave very good results in 2017. Moreover, it is still a field of research: researchers improve this architecture to make SR images more accurate. The proposed architecture is based on SRGAN, but with some changes.

First of all, batch normalization layers were removed. It helps to save computational resources and increase performance. Moreover, batch normalization produces unpleasant artifacts and limits the generalization ability, when the network is trained under a GAN framework and deeper network.

Secondly, the perceptual loss was modified: the VGG features, which are before activation layers, are used here. It makes the brightness more accurate in the reconstructed image.

Thirdly, the discriminator is improved with a relativistic discriminator. the main difference from the discriminator in SRGAN is a prediction of the probability. The probability tells us whether the real image is more realistic than a generated one, not just if it is fake or real, as it is the case in SRGAN. This modification of the discriminator allows to learn more details in image and produce sharp edges. The standard discriminator is defined as

$$D(x) = \sigma(C(x)), \tag{2.8}$$

where σ is sigmoid function, $C(x)$ is the non-transformed discriminator output. However, the relativistic discriminator can be expressed as

$$D_{Ra}(x_r, x_f) = \sigma(C(x_r) - \mathbb{E}_{x_f}[C(x_f)]), \tag{2.9}$$

where x_r – image real, x_f – image fake, $\mathbb{E}_{x_f}$ – operation of taking average for all fake data in mini-batch.

Also the Residual-in-Residual Dense Block was proposed. It combines dense connections and multi-level residual network. RRDB has more complex structure than SRGAN. The architecture is shown in Figure 2.6.

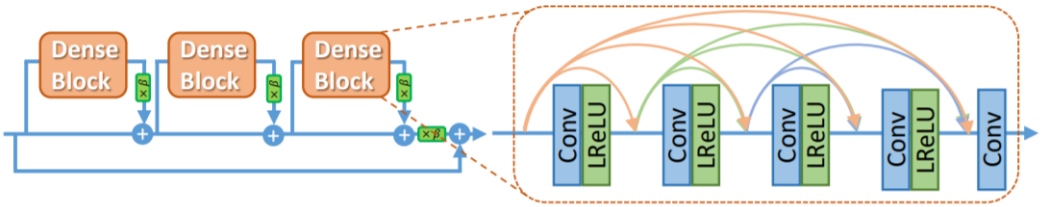


Fig. 2.6: Residual-in-Residual Dense Block. Source [17].

Another modification relates to remove noise in GAN-based models – network interpolation. Firstly, the PSNR-oriented GAN is trained and then GAN-based network is obtained by fine-tunning. It allows to produce results without artefacts and to balance perceptual quality without re-training model [17].

Progressive Face Super-Resolution via Attention to Facial Landmark

It's worth to mention this approach, which was proposed in 2019 to upscale images with faces. It upscales images with scale factor 8 and combines different approaches from super-resolution task.

First of all, this model is based on GAN architecture. The generator consists of three residual blocks. Each residual block has these layers: convolutional layer, batch normalization layer, ReLU as the activation function, transposed convolutional layer. The discriminator network has similar structure, but residual blocks has convolutional layer, Leaky ReLU as activation function, Average Pooling layer. The architecture is shown in Figure 2.7.

The architecture employs a progressive method for upscaling image. In first step, only one residual block is used in the generator. The output of this step is upscaled $\times 2$ and goes through the corresponding part of the discriminator. In the next step, the output of the previous step is upscaled also $\times 2$ by nearest-neighbor interpolation and added to the output of second residual block. The result is compared to the corresponding target images using the discriminator. The third step is similar to the second step, but the output image is scaled $\times 8$ relatively to input image.

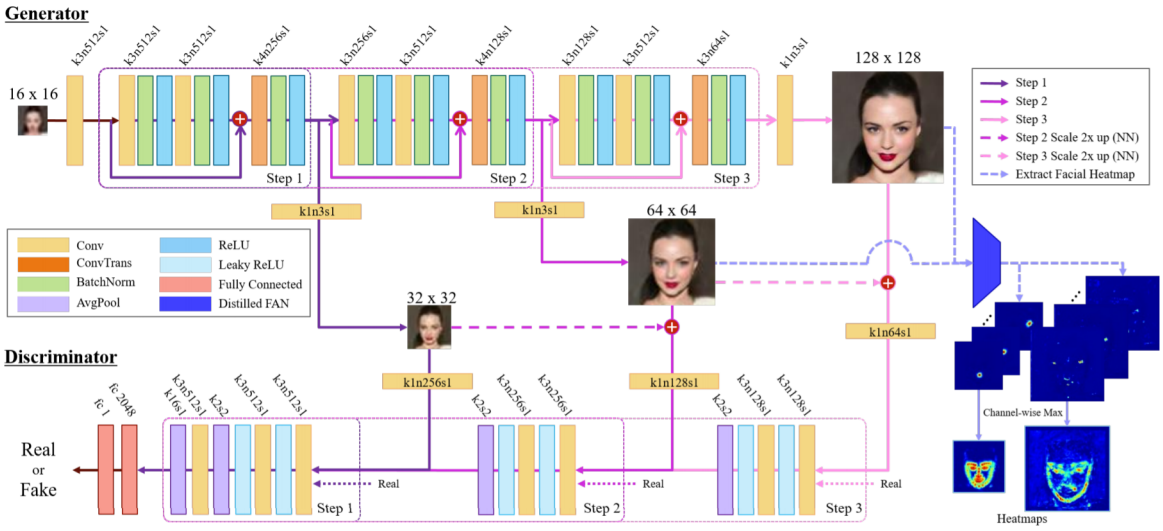


Fig. 2.7: Architecture of progressive Face SR Network. Source [32].

This approach also introduces new facial attention loss, which allows to restore facial landmarks and focuses on the facial details [32]. The facial attention loss is defined as:

$$L_{attention} = \frac{1}{r^2WH} \sum_{x=1}^{rW} \sum_{y=1}^{rH} (M^*_{x,y} \cdot |I^{HR}_{x,y} - G(I^{LR})_{x,y}|) \tag{2.10}$$

where

- M^* – heatmap,
- r – upscaling factor,
- W, H – width and height of input image,
- I^{HR} – target face image,
- I^{LR} – input low-resolution image.

The encoder-decoder network was constructed to produce heatmaps. It utilizes the state-of-the-art Face Alignment Network (FAN) [33], which predicts the location of the landmarks. In this way, it helps to minimize the error in prediction of heatmaps. This approach outperforms existing approaches.

2.2.2 Datasets

One of the important part of machine learning, is to have the suitable dataset, which the model is trained on. The most approaches used datasets DIV2K, Urban 100, BSD100 and CelebA for evaluation of models. The mentioned datasets are described below.

DIV2K

DIV2K is meant for research purposes. It contains 1000 color RGB images from the Internet and has 2K resolution images: they have 2K pixels on at least one of the axes [34]. Dataset contains training data – 800 high resolution images, validation data – 100 high resolution images and testing data – 100 high resolution images.

Urban 100

The dataset Urban 100 contains 100 HR images with a different real-world structures. Dataset is constructed with images from Flickr (under CC license) using keywords such as urban, city, architecture, and structure [35].

BSD100

A database contains ground truth segmentation for images of natural scenes, which are created by humans. This dataset can be applied for the following purposes: evaluation of performance of the segmentation algorithms and measure of probability distributions associated with Gestalt grouping factors as well as statistics of image region properties [36].

CelebFaces Attributes Dataset (CelebA)

This dataset is a large-scale face attributes dataset. It contains more than 200 000 celebrity images, each with 40 attribute annotations. CelebA has over 10 000 number of identities, over 200 000 number of face images. This dataset is suitable for face attribute recognition, face detection and landmark localization [37].

2.3 Multi-frame Super-resolution

Multi-frame super-resolution is a technique to reconstruct an image using the sequence of images of the same scene. A couple of images contains more information about the scene, which can be useful for prediction SR image. The neural networks, which can approximate complex nonlinear functions, are mostly used for single super-resolution or video super-resolution. A lot of techniques for multi-frame super-resolution focus on the restoration of the image from a sequence of the frames without using neural networks. The upscaling of the image consists of these steps: registration, fusion and reconstruction.

Registration is used to align the images into the same position, to estimate the motion information from a low resolution image and to calculate transformation

parameters. It is important to make this step correctly, because it can avoid appearing artefacts and it can also give more information about the scene. However, these methods have some limitations, such as non-uniqueness of solutions, the consideration of only translational and rotational motion between low-resolution images.

After registration the image fusion phase is used to gather information from all low resolution frames into a single image and interpolate the composed image into the high resolution grid.

And finally, in a reconstruction step the final super-resolution image is restored avoiding any distortions [38].

2.3.1 Regularization-based Approaches

Regularization is the technique of adding information to avoid overfitting and to solve ill-posed problem, including image super-resolution. The regularization based approaches can be categorized into stochastic and deterministic approaches.

Stochastic Approaches

One of the most famous method in this field is Total Variation method (TV), which was originally used in image denoising. It's very attractive for researchers, because it is able to preserve edge and detailed information. However, this method has a disadvantage: the "staircase effects" are produced in the flat regions. The effect can be reduced, but the edge and texture will be blurred [39].

One of the modifications of TV method is low-rank and total variation regularization. The main idea is to utilize the local and global information from image for more effective image reconstruction instead of only from the local neighborhoods [40]. However, this method has a heavy computational cost relative to the TV method.

Deterministic Approaches

Deterministic approaches transform the ill-posed problem into well-posed by choosing variable to minimize the Lagrangian and to solve the inverse problem by using the prior information about the solution [38].

One of the methods proposed in [41] uses the combination of L_1 norm and L_2 norm with channel weight parameters as fidelity term and adopting the regional adaptive weight coefficients as regularization term [41]. This method keeps the information about edges and the smoothness of image regions but constructed images still have the noise.

One of the successful approaches is based on Iteratively Re-weighted Least Squares (IRLS), which can be used to minimize robust m-estimators. But it is limited to only one type of robust function. The new approach of IRLS minimizes an objective function. It is composed of a data fitting function, which minimizes the error, and a set of regularization functions, which adds smoothness, edge preservation, etc. This method can be used for multi-frame super-resolution, denoising, optical flow and image blurring [42].

2.3.2 Interpolation-based Approaches

The interpolation-based approaches are the intuitive techniques for image super-resolution tasks. Mainly, these methods have following steps:

1. **Registration of images.** It is the process of alignment of selected low-resolution images to reference image. It's important to correctly approximate the movement parameters, in another case there will appear artefacts.
2. **Interpolation.** The generated high resolution image is based on estimation of new pixels from a group of pixels.
3. **Restoration.** The improvement of the reconstructed image is done after the interpolation step [43].

The simplest approach in this field is the nearest neighbor interpolation. The concept is to estimate the pixel with the value, which is the most equal among the four directly surrounded instead of the shortest distance. This method has the high performance, but produces images with a block visibility [44].

Another simple method is a bilinear interpolation. The bilinear interpolation uses the nearest 2×2 neighbors of the known pixel values around the unidentified pixel. To get its last interpolated value, a weighted average of the four pixels is computed. This approach makes images much smoother than the nearest neighbor interpolation approach. However, this approach is more complicated and has a high computation cost than the nearest neighbor interpolation approach.

Another algorithm based on the hybrid interpolation is proposed to overcome the limitations on the computational complexity and the missing details in generated image. The main idea of the hybrid image reconstruction algorithm is to separate the edge and smooth areas by the edge detector. After that the smooth area is processed with the linear interpolation algorithm, and the edge area is processed with the max relativity edge interpolation. These algorithms are processing data in parallel. The algorithm is effective for detail preserving and has a higher performance than the bicubic interpolation algorithm [45].

2.3.3 Image Registration

Thanks to the ability of deep learning to automatically learn to aggregate the information, it is possible to apply deep learning for image registration. For example, one of the method is based on reinforcement learning: agents predict small steps of transformations towards optimal alignment [46]. In this method, the registration task is separated into a sequence of classification problems, for example, to improve the alignment needs, to find the best action among a limited set of possible solutions. After repeating this process, the solution will be find. Moreover, this method has effective data augmentation, which allows to train the model on a small dataset.

Another registration method is based on fully convolutional neural networks. A registration algorithm can be optimized for a certain class of images. The proposed method estimates a transformation model for two input images directly from these images. That makes the algorithm very fast. The network is based on VGG architecture, but it is simplified, and the inputs are two three-dimensional images instead of single image [47].

2.3.4 Image Warping

Image warping is the process of manipulating in image, where the shapes were significantly distorted. This image transformation can be used for making creative images or for correcting existing deformations. Moreover, the warping can be used in super-resolution task for alignment input frames with the central frame [48]. The methods for warping can be categorized into data-scattering based and data-gathering based.

One of the data-scattering based techniques is point splatting, which is widely used in the context of image warping. However, it has the problem with increasing image resolution, because each pixel is transformed separately. That's the reason to warp such areas as blocks. This category of warping allows to work very quickly for opaque surfaces.

In data-gathering based approaches suitable color information is searched by iterations and restored from the source images. Contributing pixels are collected and composited in one step. Also, the method was enhanced by using adaptive grid warping to find suitable iteration starting points. This category of warping is suitable to composite semi-transparent image information [49].

2.4 Video Super-resolution

Video super-resolution (VSR) is also challenging problem in computer vision. Unlike single image super-resolution, VSR methods utilize information from a sequence

of images to reconstruct the frame. The most algorithms, which are originally used for single SR can't be applied to video super-resolution, at least without any modifications, because of time complexity. That was the reason for adaptation of existing architectures to video super-resolution.

2.4.1 Frame-recurrent Video Super-resolution

The Frame-recurrent Video Super-Resolution (FRVSR) is end-to-end trainable system, which allows to increase the resolution of video. One of the advantage of this system is that it doesn't estimate each frame separately, but it uses the previously estimated high resolution frame as an input for the current iteration. The architecture features the following steps to produce super-resolution.

Firstly, it makes flow estimation between the low-resolution inputs of the current frame and the previous one by FNet. It assigns a location in the low-resolution image of the previous frame to each pixel position in the low-resolution image of the current frame. Second step is upscaling the estimated flow map from the previous step using bilinear interpolation. Third step is warping of previously estimated image onto the current frame according to the optical flow from the previous frame. The fourth step is mapping of warped image from the previous step to a low-resolution space using space-to-depth transformation. The last step is super-resolution. Firstly, the low-resolution mapping of the warped output from the step 4 and current low resolution input frame are concatenated and fed to the super-resolution network SRNet.

Since this algorithm has the recurrent nature, it allows to use a large number of previous frames without increasing computational demands [50]. The described algorithm is shown in Figure 2.8.

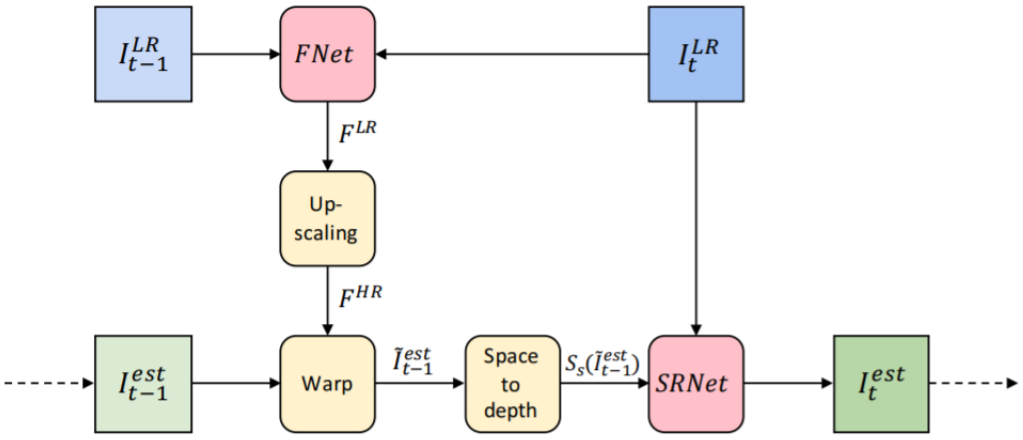


Fig. 2.8: FRVSR architecture. Source [50].

2.4.2 Temporally Coherent GANs for Video Super-resolution

The Temporally Coherent GANs for Video Super-Resolution (TecoGAN) is based on GAN and has three parts: a recurrent generator, a flow estimation network and a spatio-temporal discriminator.

The aim of the generator is to produce a high-resolution image g_t using previously generated high-resolution image g_{t-1} and a low-resolution frame x_t.

Another part of TecoGAN is network for motion estimation between x_t and x_{t-1}. The estimated motion can be resized and used as a motion compensation for a frame g_{t-1}. The architecture of the generator is shown in the Figure 2.9. To fool the spatio-temporal discriminator the generator and the flow estimator are trained together.

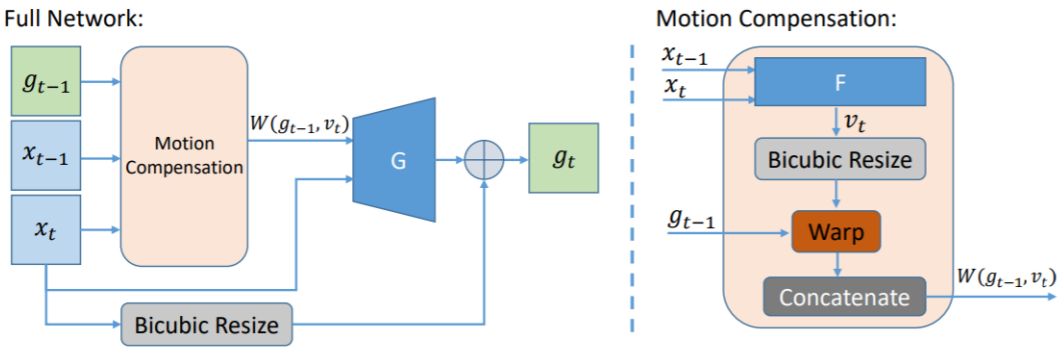


Fig. 2.9: Generator of TecoGAN. Source [51].

The last part of architecture is the discriminator. It receives ground truth set of inputs and generated set of inputs. Both sets have three adjacent high resolution frames, three corresponding low resolution frames with bicubic upsampling, and three warped high resolution frames. The inputs to the discriminator are shown in Figure 2.10.

If generated inputs have less spatial details or unrealistic artefacts compared to the real images, the discriminator will penalize the generator. After motion estimation for real and generated data, it is easier for the discriminator to define realistic and generated inputs. Thanks to the original high resolution images, it is possible to fall back to original inputs in case of unreliable motion estimation. Taking in consideration spatial and temporal inputs, the discriminator balances both aspects automatically. That allows to avoid inconsistent sharpness and overly smooth results.

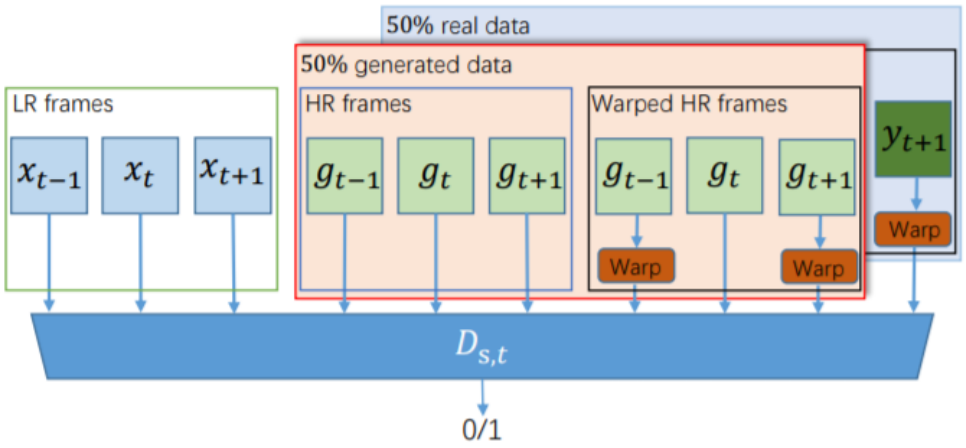


Fig. 2.10: Discriminator of TecoGAN. Source [51].

2.4.3 Enhanced Deformable Convolutional Networks

Enhanced Deformable Convolutional Networks (EDVR) is a framework, which can be applied to different image restoration tasks, for example, super-resolution and deblurring. The main cores of this architecture are an alignment module – Pyramid, Cascading and Deformable convolutions (PCD), and a fusion module – Temporal and Spatial Attention (TSA). The architecture is shown in Figure 2.11.

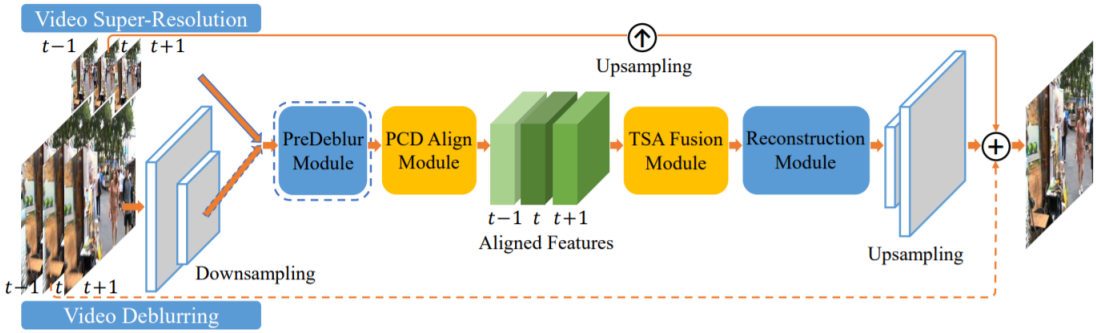


Fig. 2.11: Architecture of EDVR. Source [52].

The PCD is used to align neighbor frames to a reference frame using deformable convolutions at a feature level. The pyramid structure is used to align the features firstly in low scales and then apply the offsets and aligned features to the higher scales.

The TSA is a fusion module, which helps to aggregate information across aligned features and assign pixel-level aggregation weights on each frame. Moreover, to exploit the spatial information effectively, each location in every channel was assigned with weights too.

The framework operates with the following algorithm.

1. Input $2N + 1$ low-resolution frames are taken and high-resolution images are generated.
2. Deblurring of inputs is done with PreDeblur module to improve alignment accuracy.
3. Neighboring frames are aligned to the reference frame using PCD module.
4. The information from the different frames is fused with TSA module.
5. The fused features pass through the reconstruction module, which is a cascade of residual blocks.
6. The upsampling layer resizes the features to original input resolution.

The most operations are done in a low-resolution space, because upsampling is done at the end of the network. It saves the computational cost [52].

2.5 Application

2.5.1 Regular Video Information Enhancement

In recent years, television screen is becoming thinner and larger. The old technologies with low resolution, which were used in standard television, make the image blurred after stretching to display on these new television screens. To solve this problem, super-resolution technology is used to convert the LR images to clear HR images [53].

2.5.2 Medical Imaging

Nowadays, image super-resolution is very important in medical imaging. Various medical imaging modalities can provide both anatomical information about human body structure and functional information. However, resolution limitations always degrade the value of medical images in the diagnosis. SR technologies have been used with the key medical imaging modalities, including magnetic resonance imaging (MRI), functional MRI (fMRI), and positron emission tomography (PET) [54].

2.5.3 Biometric Information Identification

The biometric identification is actively used in security issues. There is no doubts, a lot of systems use the machine learning based core, for example, face detection and recognition needs some neural networks, since they're successful in this field. On the other hand those technologies need improvements for more accurate predictions. For face recognition it's also necessary to get an image with a good quality. It can be complicated, because the environment conditions can impact the image

from camera. That's why the super-resolution task takes part in this field of study. The aim is to make the quality of image better and to improve the cognition accuracy of classifiers by exploiting the specific characteristics of the observed biometric traits [55].

2.5.4 Security Cameras

In recent times, the networked 4K security camera systems, which record videos onto servers, have become more available. On the one hand, high resolution security camera systems are becoming increasingly convenient and widely used for monitoring areas. On the another hand, they still aren't able to produce video with a good quality in low light conditions. During the night, lighting conditions are insufficient and security cameras record low resolution and low contrast images. In situation, when the accident is done in the night, such records can be the key to find the culprit, but the problem emerges with the quality of video. That's the field of study in computer vision: how to reconstruct LR image or a couple of LR images into the one HR image, that the reconstructed image will be useful [56].

3 Implementation

The main goals of this work are to implement the architectures, which will be able to reconstruct an image containing the face from a sequence of the frames, and to compare them with methods used for single image super-resolution. Also, it is necessary to create a dataset to train the designed neural networks. The architectures should be able to reproduce faces from images, which are not recognizable by humans, for example, if recorded frames are obtained from a camera with a low grade quality. Such enhanced images can be used in law-enforcement investigation. For example, in case of the accident, when the police obtained the recording from a monitoring camera, but its quality doesn't allow to recognize the person with a naked eye. Another requirement of this work is to utilize Python language for implementation of neural networks. Python offers a variety of libraries, which can be used in machine learning. One of such libraries is Keras, which has simple syntax and allows the programmers to focus on the model structure instead of implementation of each layer. In this work, Keras is used to create the neural models.

Tensorflow is used as a backend for Keras. It is a math library, which provides an effective computation, that makes this library suitable for the machine learning tasks. Moreover, Tensorflow supports Graphics Processing Unit (GPU), which allows to train models much more faster than with traditional Central Processing Unit (CPU).

For this work the dataset has been created and rescaled to experiment with different scale factors. The description of the dataset is in Section 3.1.

The next step was to train the existing methods for single super-resolution. The chosen models are well known state-of-the-art for super-resolution task and they were described in Section 2.2. All source codes are available on the Internet: SRCNN[1], SRGAN[2], EDSR[3], ESRGAN[4]. Other mentioned methods in that section, such as Progressive Face Super-Resolution via Attention to Facial Landmark and Edge Enhanced Single Image Super-resolution, don't have published training codes. The models were adapted to train on a new dataset and also to train with different scales.

After that, the new architectures for multi-frame super-resolution were proposed. The description of them is in Section 3.2. The training of models was performed on GPU NVidia Tesla P100-PCIE-16GB and NVidia GTX 1080Ti-12GB.

[1] https://github.com/MarkPrecursor/SRCNN-keras
[2] https://github.com/tensorlayer/srgan
[3] https://github.com/jmiller656/EDSR-Tensorflow
[4] https://github.com/open-mmlab/mmsr

3.1 Dataset

As it was mentioned above, there is a need to create a dataset for training and testing of proposed architectures. Movie trailers from the Youtube were used to create the dataset. Firstly, the cropped frames with faces were created with an architecture based on Single Shot MultiBox Detector (SSD). This architecture was pretrained to detect faces in images. Then the sequences of 6 consecutive frames were chosen, while paying attention to the size of images, which should have a minimum size 256×256 px. This size allows to downscale images with 2, 4 and 8 scale factors. The next step was to divide the sequences of the images into training, validation and testing parts. The training part contains 280 sequences, the validation part has 70 sequences in it and the testing part is composed of 34 sequences. All frames were resized to 32×32 px. A label is the fourth frame, which has the size 64×64 px for $\times 2$ scale, 128×128 px for $\times 4$ scale and 256×256 px for $\times 8$ scale.

The dataset is separated into the directories by number of the image in the sequence: all first frames are in first directory, the second frames are in the second directory and etc. This method of a separation allows to push images in the correct order to the application. Also there's a directory with the label images. The example of the input and the output images is in Figure 3.1

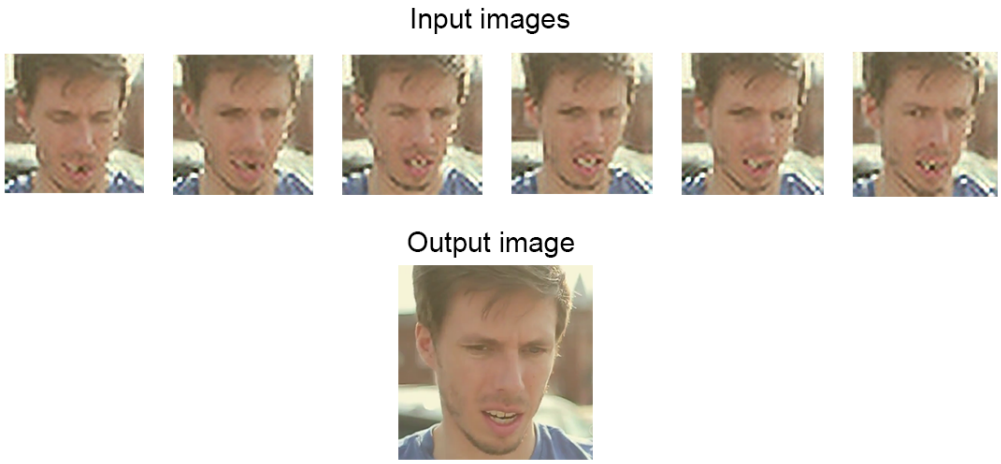


Fig. 3.1: Example of the input and the output in training dataset.

3.2 Description of Implemented Models

For this experiment, the U-Net model was chosen and modified to produce upscaled images. The original model is described in Section 1.5. The training was run with

500 epochs and 500 steps per epoch. The chosen batch size is 4. To make the training easier, there's a main file, which launches the training. It contains the general parameters for all models, for example, the input and output size, a scale factor, a path for weights, paths for training and validation data. Each model has its own directory contained a file with the implemented architecture. It helps to manage the files created during training and testing.

To get images from directories, Keras function `flow_from_directory` is used. Since the input contains 6 frames, this function is applied six times. The label images are also taken with function `flow_from_directory`. This preprocessing is placed in separate file `data.py`.

To train the model the Keras function `fit_generator` is used. It allows to setup different parameters, such as the number of epochs, step per epoch, training and validation data, and to start the training.

3.2.1 U-Net with Residual blocks

The first proposed architecture is based on the U-Net model and the residual blocks. Before the input sequence of images is fed into the network, the frames are upscaled to desired size with the bicubic interpolation. Each upscaled frame is applied to the combination of convolutional and LeakyReLU layers, and this is repeated 3 times. This operation helps to extract feature maps from input frames. Then all these feature maps are concatenated. After this step, 8 residual blocks are employed. The output of the residual blocks is added to the extracted feature maps from the fourth frame. And finally the U-Net model is applied. A contracting path has blocks composed of combination of the convolutional layer and the ReLU repeated twice, and of the pooling layer.

At the beginning of U-Net model the convolutional layers start with extracting 32 feature maps and with every block the number of them is doubled. The bottleneck has 1024 feature maps and size of them is 8×8. The main difference from the original model is application of Subpixel convolutional layer [57] instead of standard Upsampling layer in the expanding path. Subpixel convolution is often used in super-resolution tasks and achieves better results. The output of U-Net is a resulting image. The used loss function is MSE. This loss function allows to get higher results for PSNR values, however, the models with it are not able to recover details and textures [58], consequently, the output images are blurred. The architecture is shown in Figure 3.2 and named as "U-Net + ResBlocks" in the tables with the results.

3.2.2 U-Net with GEU blocks

Another method for upscaling is used in next proposed architecture: upscaling is not used in pre-processing as in the previous model, but during the training using a weighted layer – Subpixel convolutional layer.

Input is a sequence of 6 frames with size 32×32 px, which are concatenated. After that the stack of 5 GEU [59] blocks is used. This block is based on a residual block. In spite of good performance, the authors of GEU modified the residual block, so that input and output are not just simply added, but they are added by learned weight values. It has the next structure: dropout layer, 3 convolutional layers, concatenate layer, which adds the output of convolutional layer to input layer received by the block, convolutional layer, sigmoid activation layer and Union layer. Union layer performs the next operation: $g \times y + (1 - g) \times x$, where x – input layer, y – third convolutional layer, g – sigmoid activation layer.

After these blocks, there are upscaling layers – Subpixel convolutional layers. Each layer increases the size of the feature maps by $\times 2$. The U-Net model is applied after the upscaling layers. The difference between original one and the proposed one is the utilization of the Sub-pixel convolutional layers and the Batch normalization layers in the expanding path.

Moreover, instead of using the standard MSE loss function, the VGG19-based loss function is used. The used feature reconstruction loss [60] computes a loss of pretrained network, usually of VGG16 or VGG19 network. The main difference between the pixel loss functions, such as MSE, and VGG19-based loss functions is the computation of loss between feature representations instead of generated and target images. This kind of loss function can avoid blurring and is able to reconstruct more details, however, the output images suffer from the artefacts. The architecture is shown in Figure 3.3 and is named as "U-Net + GEU" in the tables with the results. To reduce the artefacts in output images, the Gaussian filter has been applied. The kernel size varies for different output image sizes: for output with the size of 64×64 px the kernel size is 3, for the size 128×128 px the kernel size is 5, for the size 256×256 px the kernel size is 9.

3.2.3 U-Net with GEU blocks – improvements

The described model in Subsection 3.2.2 produces sharper image, than the model, which is described in Subsection 3.2.1. However, the combination of VGG19-based loss function and Sub-pixel convolutional layers in the U-Net model produces artefacts, called CheckerBoard artefacts [61]. To reduce them, the used the U-Net model is the original one: with the Upsampling layers, without the Batch Normalization

layers. Moreover, the Convolutional layers at the beginning of the proposed architecture has the kernel size of 7. It allows the network to see a larger area and to extract larger features from a face. Also the GEU block is modified: the original block uses the Sigmoid activation function in the end, which was also used in Subsection 3.2.2, but it is replaced with the ReLU activation function in the new model. The used loss function is a feature reconstruction loss, as it's in the previous model. The mentioned improvements are shown in Figure 3.4 and named "U-Net + GEU 2" in the tables with the results.

Another implemented modification is replacing the GEU blocks between the Subxel convolutional layers and the U-Net model. It allows to extract more features from the upscaled inputs and to take more information about the scene. The features are extracted by the Convolutional layers with the kernel of size 7. The U-Net model has the same structure, as in the previous one. The U-Net model works here as a filter, which helps to reduce the artefacts.

This modification also suffers from artefacts, but they're smoother and there's a possibility, that they can disappear with a larger training dataset. The designed architecture is shown in Figure 3.5 and named "U-Net + GEU 3".

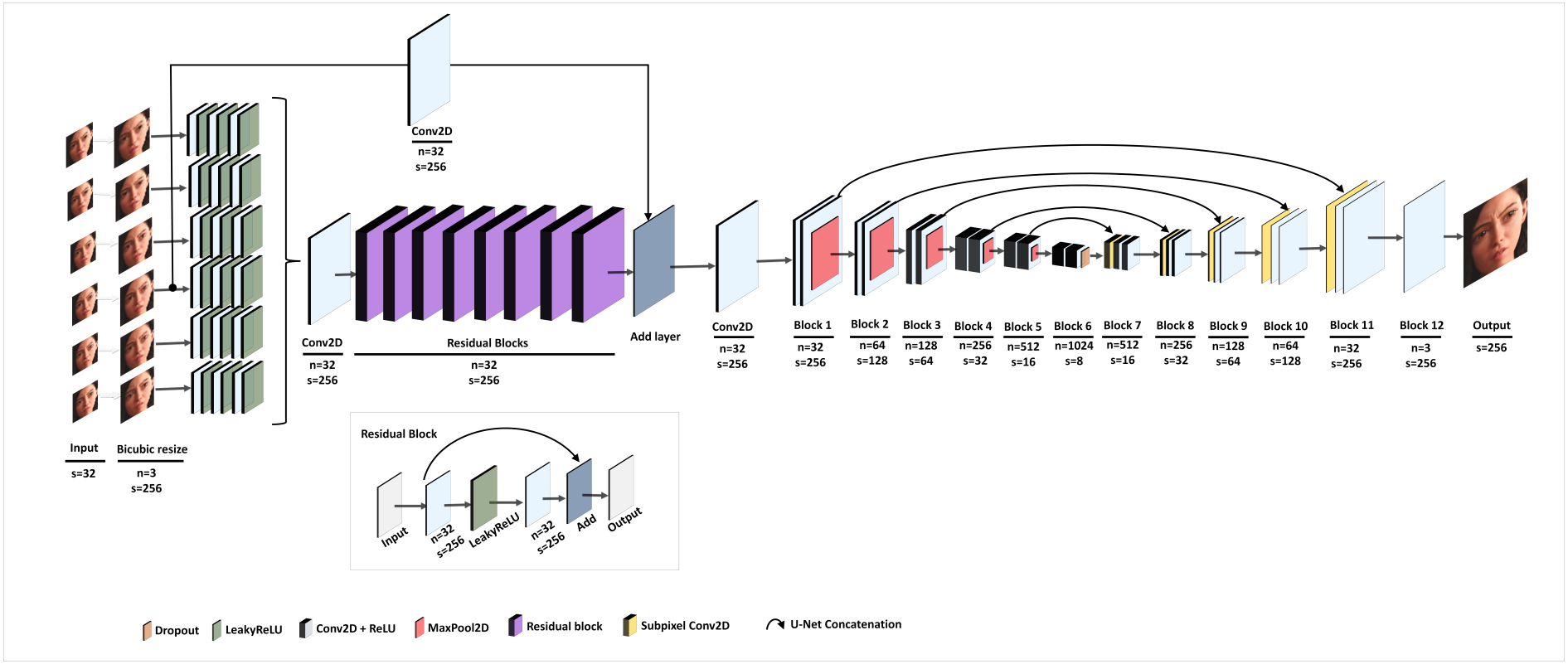


Fig. 3.2: Proposed architecture – U-Net with Residual blocks, where n – number of filters, s – size of filters

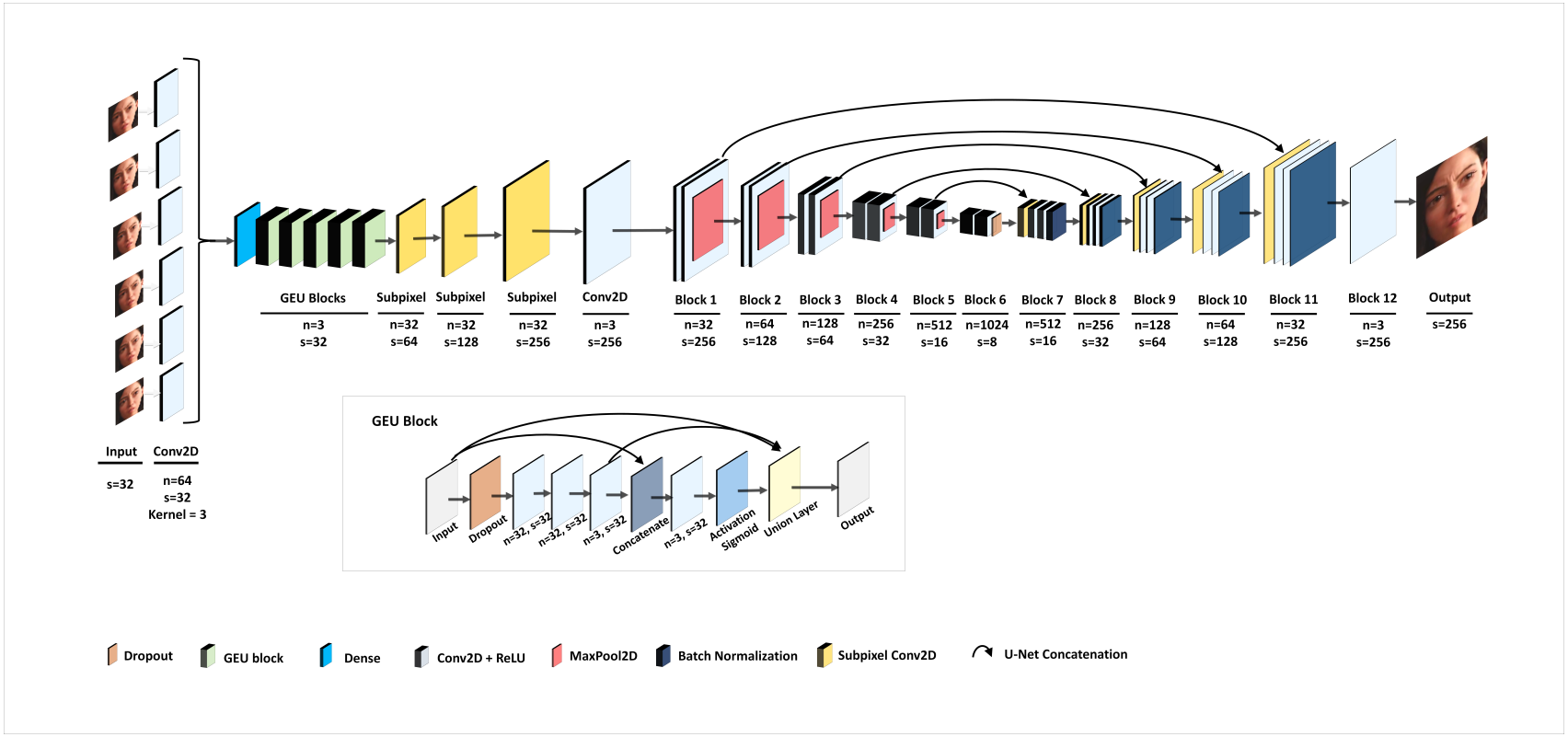


Fig. 3.3: Proposed architecture – U-Net with GEU blocks, where n – number of filters, s – size of filters

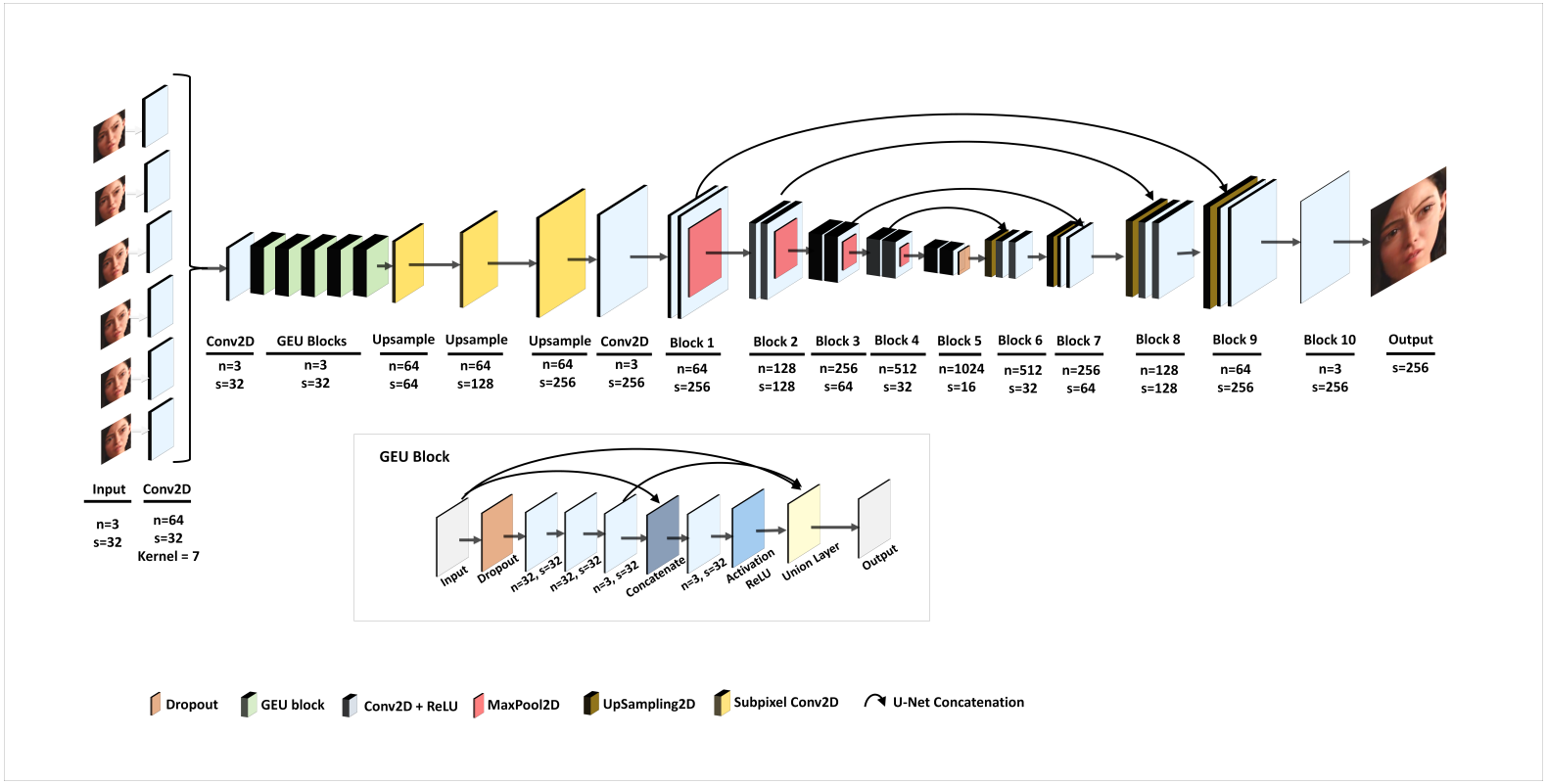


Fig. 3.4: Proposed architecture – modified U-Net with GEU blocks (U-Net + GEU 2), where n – number of filters, s – size of filters

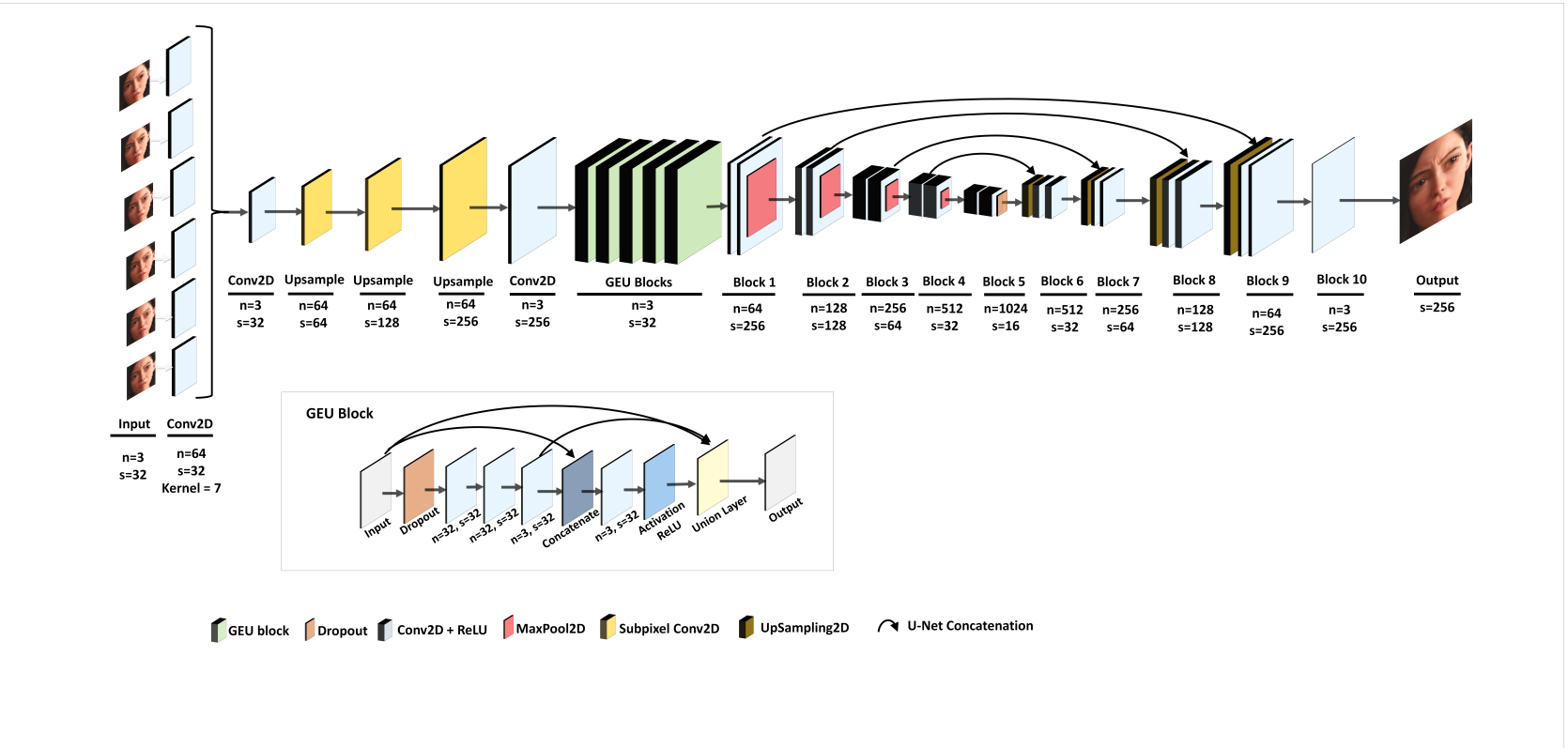


Fig. 3.5: Proposed architecture – modified U-Net with GEU blocks (U-Net + GEU 3), where n – number of filters, s – size of filters

3.3 Used Framework

3.3.1 Keras

Keras[5] is a top-level API for Tensorflow and Theano. It allows to define the neural network very easily while the developer can focus on architecture, but not on implementing each layer, that makes this library very user friendly. Keras can be run on GPU and CPU. It provides the understandable feedback upon user error. Another point is modularity. Keras allows to combine different modules, such as neural layers, cost functions, activation functions and other, and to plug them together with minimal restrictions. Moreover, the new models can be added for the advanced research. Installation of Keras requires Tensorflow, Theano or CNTK. If GPU is used, it is also necessary to install CuDNN and CUDA driver.

3.3.2 Tensorflow

Tensorflow [6] is an open-source library for numerical computations, including machine learning. It offers multilevel API. For high-level API can be used, for example, Keras, which allows to build the model very fast. However, if there's a need to train model with different computational resources, for example, with some different GPUs, the Distribution Strategy API can be used. Tensorflow can be used with such programming languages, as C++, Java, JavaScript, Python and can be run on CPU, GPU and TPU.

[5]https://keras.io/

[6]https://www.tensorflow.org/

4 Results

The focus of this work is restoration of the face images, which are not recognizable by humans. The results of this work are composed by the created dataset with the sequences of the images, the comparison of the implemented architectures for multi-frame super-resolution and existing methods for single frame super-resolution. The models were trained on the dataset, as described in Section 3.1.

4.1 Metrics

The objective metrics, which are used for evaluation models, are: Structural similarity (SSIM), Peak signal-to-noise ratio (PSNR), Mean squared error (MSE), sharpness and face recognition.

4.1.1 Mean Squared Error (MSE)

MSE measures the average of the squares of the difference between the actual value and the estimated values. MSE is computed as [62]

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - X_i)^2, \tag{4.1}$$

where

- X_i – high-resolution image,
- Y_i – super-resolution image,
- n – training examples.

4.1.2 Peak Signal-to-Noise Ratio (PSNR)

PSNR allows to compare image compression quality in decibels [dB]. It measures the difference between the maximal possible signal and noise-affected one. The usual range of PSNR value is 20 - 40 dB. As higher the value of PSNR, than the quality of image is better. The PSNR can be defined as

$$PSNR = 10 \cdot \log(\frac{MAX^2}{MSE}), \tag{4.2}$$

where

- MAX – the maximum pixel value,
- MSE – the Mean Squared Error.

4.1.3 Structural Similarity (SSIM)

SSIM is designed to improve the traditional metrics PSNR and MSE. Image distortion is modeled as a combination of three factors, which are loss of correlation, luminance distortion and contrast distortion. SSIM can be computed as [63]:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1) + (2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{4.3}$$

where

- μ_x and μ_x – the average of images x and y,
- μ_x^2 and μ_x^2 – the variance of images x and y,
- σ_{xy} – the covariance of images x and y,
- C_1 and C_2 – two variables, which depend on range of pixel-value.

4.1.4 Sharpness (CPBD)

It is a no-reference objective sharpness metric based on a cumulative probability of the blur detection [64]. CPBD is based on the study of the human blur perception for varying contrast values.

The metric uses a probabilistic model to predict the probability of detecting blur at each edge in the image, after that the cumulative probability of blur detection (CPBD) is computed, as follows:

$$CPBD = P(P_{BLUR} \leq P_{JNB}) = \sum_{P_{BLUR}=0}^{P_{BLUR}=P_{JNB}} P(P_{BLUR}), \tag{4.4}$$

$$P_{BLUR} = P(e_i) = 1 - exp(-\left|\frac{w(e_i)}{w_{JNB}(e_i)}\right|^{\beta}), \tag{4.5}$$

where

- $w(e_i)$ – measured width of the edge e_i,
- $w_{JNB}(e_i)$ – "Just Noticeable Blur" (JNB) edge width, which depends on the local contrast around edge,
- β – parameter, which is obtained after applying curve fitting.

4.1.5 Face Recognition

Face recognition is a comparison of a given list of face encodings and a known face encoding. This metric is based on a computed euclidean distance for each comparison face. The distance defines, how similar the faces are, so the less the value of distance, the more similar the faces are[1].

[1]https://face-recognition.readthedocs.io/en/latest/readme.html

4.1.6 Subjective metrics

Sometimes the results of objective metrics and human perception are different. For example, when PSNR and MSE show relatively good results, a real person can see, that image is blurred or the details are deformed. That's why, it was decided to ask for the opinion of a group of people – which output image looks better and more similar to the ground truth image than others. For that aim, the 19 people were chosen to vote. The used images are shown in Figure 4.1, Figure 4.2 and Figure 4.3. In the table, there's a percentage, showing how many people think that the method looks better more similar to the original one.

4.2 Evaluation of single super-resolution models

As it was mentioned, the chosen models are SRCNN, EDSR, SRGAN, ESRGAN. They were trained on the new dataset. For 2 scale factor the SRCNN has higher SSIM, MSE, PSNR values. Also it is better than standard methods, such as bilinear and bicubic interpolation. However, the output image is blurred. On the other side, the SRGAN output is sharper than other single frame methods. It can be conformed by the high value of sharpness.

For 4 and 8 scale factors SRCNN also produces better results. But it can be seen, that the image suffers from more blur, than for 2 scale factor. The number of failed face recognition is larger than it was with GAN-based models. SRGAN and ESRGAN has more face deformations, that's why faces don't look realistic.

4.3 Evaluation of implemented models

The implemented models were tested on created dataset, which was mentioned in Section 3.1. The models for single super-resolution and the proposed models for multi-frame super-resolution are evaluated using objective metrics – MSE, PSNR, SSIM, sharpness of original and predicted images, face recognition, and subjective metrics. For evaluation, the testing part of dataset is used. The average values for each model are shown in Tab. 4.1, Tab. 4.2 and Tab. 4.3. The images are shown in Figure 4.1, Figure 4.2 and Figure 4.3.

For 2 scale factor, the images were upscaled to size 64×64 px. According to the results, SRCNN has the best results of MSE, SSIM, PSNR metrics. However, the proposed method "U-Net + ResBlocks" has better MSE value, than EDSR and SRGAN methods. Moreover, SRCNN, EDSR, SRGAN, "U-Net + GEU 2", "U-Net + GEU 3" has the least number of failed face recognition. This can be important in case of identification of person. Proposed "U-Net + GEU 2" has the least difference

of sharpness value. "U-Net + GEU" has very visible artefacts, because the output size of image it still relative small. This causes large error and low PSNR value. On the other side, the "U-Net + ResBlocks" has better quantitative results. However, the output image is blurred and, as expected, the details are not recovered. However, 42% people voted for method "U-Net + GEU 3", which is the highest value in this metric.

The images with 4 scale factor have size 128×128 px. SRCNN has also better results in SSIM, MSE, PSNR. But the second place among the neural network methods has "U-Net + GEU" (with Gaussian filter). The closest sharpness value to original image has SRGAN method. These methods has the least number of failed face recognition: SRGAN, ESRGAN, "U-Net + GEU 2", "U-Net + GEU 3". From the subjective side, the SRGAN and ESRGAN methods produce some face deformation. This fact is very important for face recognition. The proposed "U-Net + GEU" method still has artefacts, but they are less visible, comparing to the output with 2 scale factor. This effect is reduced by the Gaussian filter and quantitative values are better than values of SSIM and MSE for SRGAN and EDSR. On the other hand, 68% votes has the "U-Net + GEU 3" model.

The most state-of-the-art approaches try to solve the problem for 2 and 4 scale factor. However, it's more complicated to recover image for 8 scale factor. That's why this work also try to solve the problem with 8 scale factor. The images were upscaled to size 256×256 px. From the objective side, SRCNN still has better results. The proposed "U-Net + ResBlocks" has the second place for SSIM, MSE, PSNR metrics. The proposed methods have less number of failed face recognition, than methods for single super-resolution.

From the subjective side, EDSR and SRCNN have worse recovered details: they are blurred and have artefacts. SRGAN suffers from deformation. The proposed "U-Net + ResBlocks" also has a problem with details and texture recover. In spite of good quantitative metrics, for human it's necessary to see the details of face, especially, if it is used in the identification of the person. Proposed "U-Net + GEU" has checkerboard artefacts, however, in image with larger size it looks not so fatal and the face still can be recognised. The Gaussian filter helps to make the image cleaner, but it will have the blur effect. The improvements of that model gives better result: "U-Net + GEU 2" produces image without artefact, the "U-Net + GEU 3" recovers more details, but with some artefacts. The highest value of subjective metric has the "U-Net + GEU 2".

According to the results, it can be seen, that the objective metrics can be different from subjective ones. In spite of successful SRCNN model in objective metrics, the proposed methods are better for human perception, which is confirmed with the results from questionnaire. The problem of face super-resolution requires to

reconstruct more details of face and standard methods without modification can't be simply applied to it. The models with different loss functions and with different methods for upscaling images produce much better results and allow to recover texture and details, even from images, which are not recognizable by humans.

Tab. 4.1: Results for 2 scale factor.

Model	SSIM	MSE	PSNR, dB	Sharp. SR	Sharp. origin	Sharp. difference	Face recognition	Failed face recognition	Subjective opinion, %
Bilinear	0.8952	87.3707	29.3002	0.1856	0.3957	0.2101	0.3867	11	0
Bicubic	0.9054	84.0203	29.5706	0.3211	0.3957	0.0746	0.3780	8	5.26
SRCNN	0.9321	57.6459	31.1931	0.2594	0.3957	0.1363	0.3239	7	0
EDSR	0.8716	155.2390	26.6462	0.4812	0.3957	-0.0855	0.4006	7	0
SRGAN	0.8767	120.8760	27.9672	0.5149	0.3957	-0.1196	0.4075	7	15.79
U-Net + ResBlocks	0.8713	119.051	27.9567	0.1709	0.3957	0.2248	0.4091	13	0
U-Net + GEU	0.6549	363.4217	22.8523	0.5634	0.3957	-0.1678	0.4445	8	5.26
U-Net + GEU (with gaussian filter)	0.8714	145.2156	27.0560	0.1977	0.3957	0.1980	0.3976	9	5.26
U-Net + GEU 2	0.8589	140.4150	27.1711	0.3804	0.3957	0.0153	0.4238	7	26.32
U-Net + GEU 3	0.7865	180.5470	25.9309	0.4881	0.3957	0.0925	0.4072	7	42.11

Tab. 4.2: Results for 4 scale factor.

Model	SSIM	MSE	PSNR, dB	Sharp. SR	Sharp. origin	Sharp. difference	Face recognition	Failed face recognition	Subjective opinion, %
Bilinear	0.8686	83.5592	29.4994	0.0102	0.2735	0.2634	0.4066	8	0
Bicubic	0.8769	80.8359	29.7292	0.0255	0.2735	0.2480	0.4051	7	0
SRCNN	0.8971	60.4080	30.9974	0.0311	0.2735	0.2424	0.3786	7	0
EDSR	0.8354	108.2120	28.1768	0.1706	0.2735	0.1029	0.4217	8	0
SRGAN	0.8516	88.6431	29.2538	0.2445	0.2735	0.0290	0.4223	5	0
ESRGAN	0.8753	79.3104	29.8076	0.2350	0.2735	0.0385	0.3975	5	10.53
U-Net + ResBlocks	0.8486	110.8600	28.1761	0.0320	0.2735	0.2415	0.4160	10	0
U-Net + GEU	0.4761	363.1138	22.7166	0.6500	0.2735	-0.3764	0.4213	6	0
U-Net + GEU (with Gaussian filter)	0.8635	87.7607	29.2530	0.0409	0.2735	0.2325	0.4125	6	15.79
U-Net + GEU 2	0.7213	181.5750	26.0962	0.4334	0.2735	-0.1599	0.4260	5	5.26
U-Net + GEU 3	0.7386	142.041	26.9914	0.5433	0.2735	-0.2698	0.4158	5	68.42

Tab. 4.3: Results for 8 scale factor.

Model	SSIM	MSE	PSNR, dB	Sharp. SR	Sharp. origin	Sharp. difference	Face recognition	Failed face recognition	Subjective opinion, %
Bilinear	0.8580	83.7375	29.4899	0	0.1353	0.1353	0.4155	10	0
Bicubic	0.8620	81.1447	29.7114	0	0.1353	0.1353	0.4143	8	0
SRCNN	0.8759	68.7263	30.4378	0.0199	0.1353	0.1154	0.4077	9	5.26
EDSR	0.8242	103.3580	28.3724	0.3055	0.1353	-0.1703	0.4187	8	5.26
SRGAN	0.8159	97.6540	28.8205	0.1468	0.1353	-0.0115	0.4241	8	0
U-Net + ResBlocks	0.8662	78.4192	29.8387	0.0210	0.1353	0.1143	0.4078	8	10.53
U-Net + GEU	0.4379	303.1290	23.4900	0.5727	0.1353	-0.4375	0.4266	7	0
U-Net + GEU (with Gaussian filter)	0.8514	94.7414	28.8317	0.0014	0.1353	0.1338	0.4181	7	10.53
U-Net + GEU 2	0.8252	136.3850	27.4542	0.0498	0.1353	0.0854	0.4104	8	36.84
U-Net + GEU 3	0.6634	166.0480	26.2956	0.4414	0.1353	-0.3062	0.4132	7	31.58

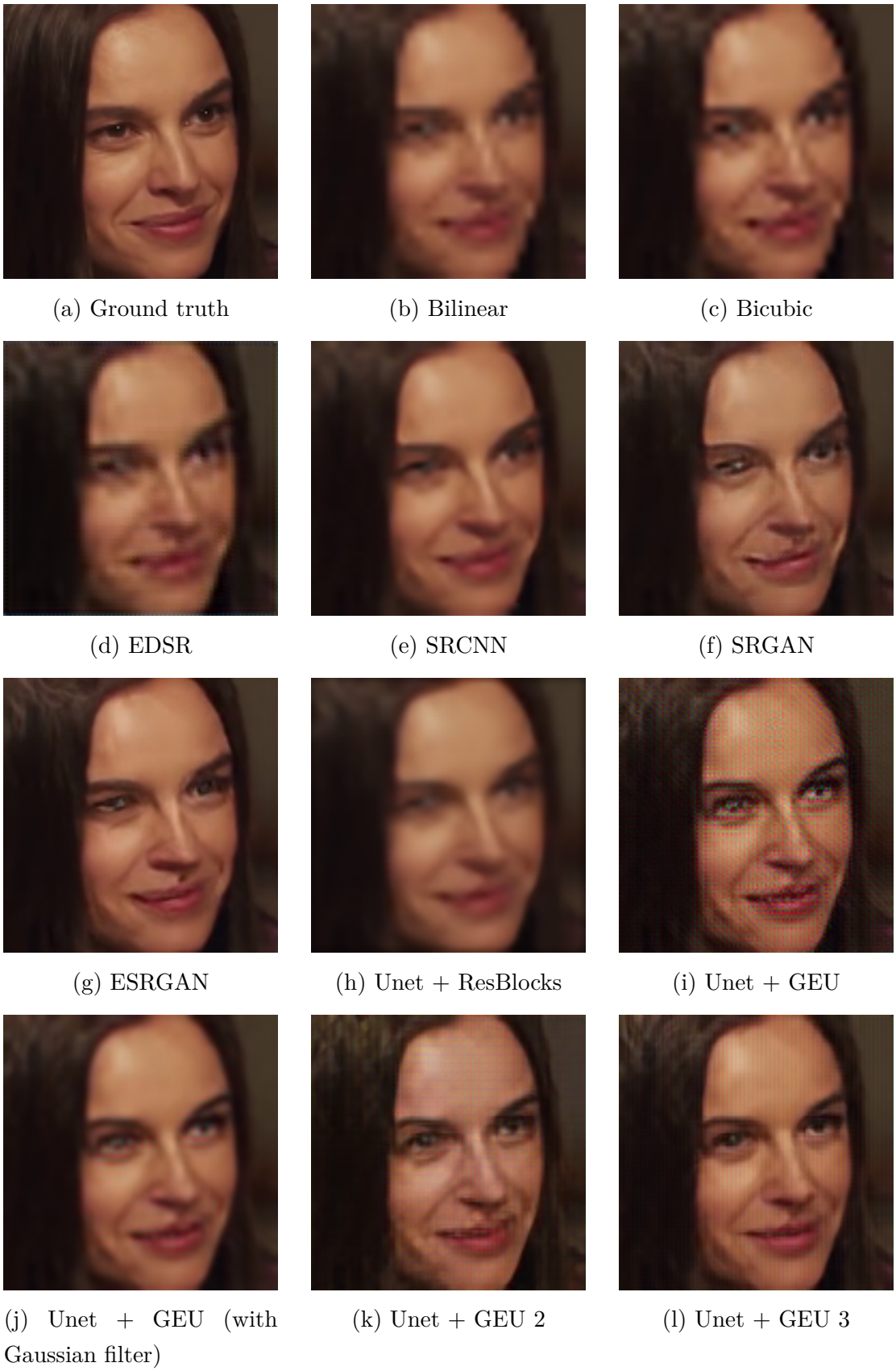
(a) Ground truth (b) Bilinear (c) Bicubic

(d) EDSR (e) SRCNN (f) SRGAN

(g) Unet + ResBlocks (h) Unet + GEU (i) Unet + GEU (with Gaussian filter)

(j) Unet + GEU 2 (k) Unet + GEU 3

Fig. 4.1: $\times 2$ upscaling.



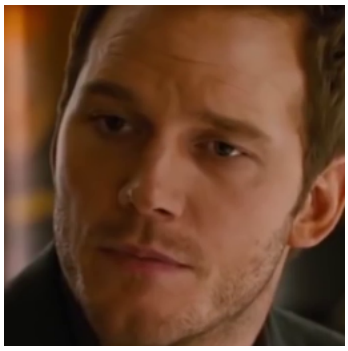
(a) Ground truth (b) Bilinear (c) Bicubic

(d) EDSR (e) SRCNN (f) SRGAN

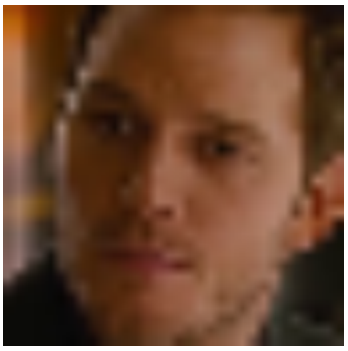
(g) ESRGAN (h) Unet + ResBlocks (i) Unet + GEU

(j) Unet + GEU (with Gaussian filter) (k) Unet + GEU 2 (l) Unet + GEU 3

Fig. 4.2: $\times 4$ upscaling.

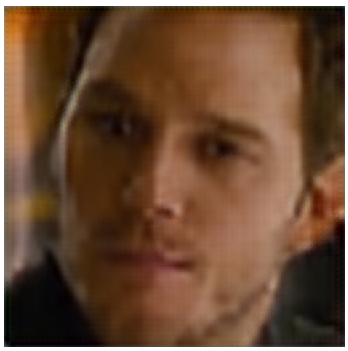


(a) Ground truth

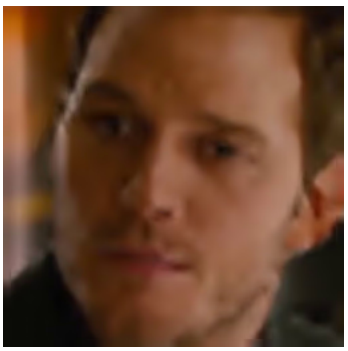


(b) Bilinear

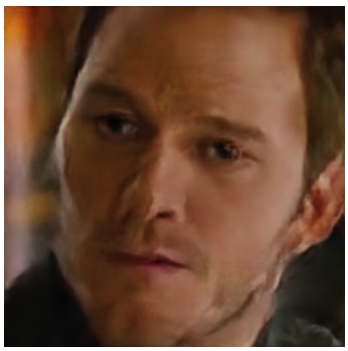
(c) Bicubic



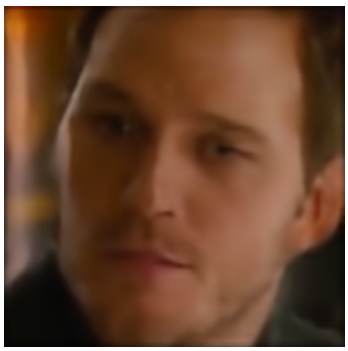
(d) EDSR



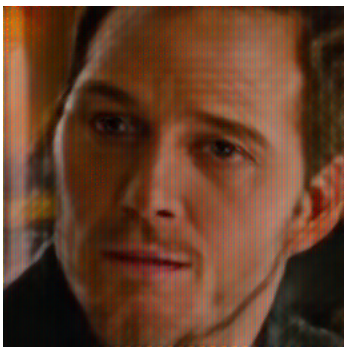
(e) SRCNN



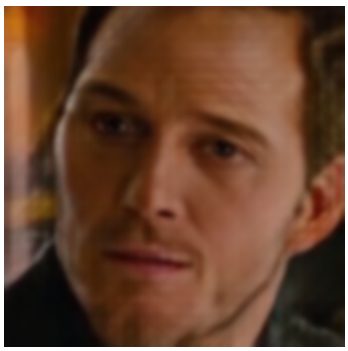
(f) SRGAN



(g) Unet + ResBlocks



(h) Unet + GEU



(i) Unet + GEU (with Gaussian filter)

(j) Unet + GEU 2

(k) Unet + GEU 3

Fig. 4.3: $\times 8$ upscaling.

5 Conclusion

The objectives of this work were to propose the architectures, which would be able to reconstruct a face image using a sequence of the frames and to compare with the methods for single image super-resolution.

Methods for single image super-resolution, face super-resolution, multi-frame super-resolution, video super-resolution and application of these methods were studied. Super-resolution can be applied for increasing resolution of images from camera, monitoring streets and buildings for detection of suspects.

In the next part of the work, the dataset with the sequences of frames with faces was created. The methods for single image super-resolution were trained on the new dataset: the fourth frame was used. After that, some architectures, which would be able to reconstruct image from a sequence of frames, were implemented. The aim was to get an upscaled image from low-resolution inputs, where the face is not recognizable by humans. For that purpose the upscaling factors 2, 4 and 8 were chosen, it means, that upscaling of image has been done from 32×32 px to 64×64 px, 128×128 px and 256×256 px respectively. As the last step, the proposed architectures were compared with the trained methods for single super-resolution.

The proposed architectures were based on U-Net model. However, different modifications were done: residual blocks and GEU blocks were added, MSE and perceptual loss functions were utilized, different methods of upscaling were used.

The main contributions of this thesis are upscaling of images, so the face can be recognized by humans and comparison methods for single images super-resolution and multi-frame super-resolution. Utilization of the GEU blocks and the perceptual loss function allows to recover more details in images, so the number of failed face recognition is reduced.

The quantitative metrics for images with $\times 2$ scale: SSIM is 0.62–0.87, MSE is 140–363, PSNR is 22–27 dB. Images with $\times 4$ scale factor have SSIM is 0.47–0.86, MSE is 87–363, PSNR is 22–29 dB. For $\times 8$ upscale factor SSIM is 0.43–0.86, MSE is 78–303, PSNR is 23–29 dB. The worst quantitative results has Unet + GEU because of the artefacts. However, they can be fixed using filters, for example, Gaussian.

From subjective side, the outputs of proposed methods with GEU blocks are sharper than outputs of single image super-resolution. Moreover, with larger scale factor the difference between quality of images is more visible.

According to these results, the implemented architectures can be applied in practice, for example, for upscaling of images from security cameras, for increasing the quality of face image, which can be after that recognized. However, there's also the place for improvements, for example, decreasing the blurring, extending dataset, reducing artefacts to get more accurate results.

Bibliography

[1] BENDETT, S. *Moscow to Weave AI Face Recognition into Its Urban Surveillance Net* [online]. 2019, last actualization 1. 5. 2019 [cit. 20. 10. 2019]. Available: <https://www.defenseone.com/technology/2019/05/moscow-weave-ai-face-recognition-its-urban-surveillance-net/156994>

[2] SIU, W.; HUNG, K. Review of image interpolation and super-resolution. In *Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference*, 2012. p. 1–10.

[3] BELL, J. Machine Learning: Hands-On for Developers and Technical Professionals. *Wiley*, 2014, ISBN 9781118889497.

[4] KARPATHY, A. *Yes you should understand backprop* [online]. 2016, [cit. 20. 10. 2019]. Available: <https://medium.com/@karpathy/yes-you-should-understand-backprop-e2f06eab496b>

[5] GUPTA, D. *Fundamentals of Deep Learning – Activation Functions and When to Use Them?* [online]. 2020, [cit. 20. 04. 2020]. Available: <https://www.analyticsvidhya.com/blog/2020/01/fundamentals-deep-learning-activation-functions-when-to-use-them/>

[6] MAAS, A.; HANNUN, A.; NG, A. Rectifier nonlinearities improve neural network acoustic models. *Proc. icml*, 2013, vol. 30, no. 1, p. 3.

[7] DESHPANDE, A. *A Beginner's Guide To Understanding Convolutional Neural Networks* [online]. 2016, [cit. 20. 03. 2020]. Available: <https://adeshpande3.github.io/adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks>

[8] KARPATHY, A. *Convolutional Neural Networks (CNNs / ConvNets)* [online]. 2020, [cit. 20. 04. 2020]. Available: <https://cs231n.github.io/convolutional-networks>

[9] IOFFE, S.; SZEGEDY, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[10] *Fully Connected Layers in Convolutional Neural Networks: The Complete Guide* [online]. 2020, [cit. 20. 04. 2020]. Available: <https://missinglink.ai/guides/convolutional-neural-networks/fully-connected-layers-convolutional-neural-networks-complete-guide>

[11] RANGANATHAN, V.; NATARAJAN, S. A new backpropagation algorithm without gradient descent. *arXiv preprint arXiv:1802.00027*, 2018.

[12] HU, X.; NAIEL, M.; WONG, A.; LAMM, M.; FIEGUTH, P. RUNet: A Robust UNet Architecture for Image Super-Resolution. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, p. 0–0.

[13] RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical image computing and computer-assisted intervention*, 2015, p. 234–241.

[14] GARG, L.; SHUKLA, P.; SINGH, S. K.; BAJPAI, V.; YADAV, U. Land Use Land Cover Classification from Satellite Imagery using mUnet: A Modified Unet Architecture, 2019.

[15] LEDIG, C.; THEIS, L.; HUSZÁR, F.; CABALLERO, J.; CUNNINGHAM, A.; ACOSTA, A.; AITKEN, A.; TEJANI, A.; TOTZ, J.; WANG, Z. AND OTHERS Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017. p. 4681–4690.

[16] LIM, B.; SON, S.; KIM, H.; NAH, S.; LEE, K. M. Enhanced Deep Residual Networks for Single Image Super-Resolution. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 06. 2017. ISBN 9781538607336

[17] WANG, X.; YU, K.; WU, S.; GU, J.; LIU, Y.; DONG, C.; QIAO, Y.; CHANGE LOY, C. Esrgan: Enhanced super-resolution generative adversarial networks. *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, p. 0–0.

[18] HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016. p. 770–778.

[19] SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[20] GOODFELLOW, I.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A.; BENGIO, Y. Generative adversarial nets. *Advances in neural information processing systems*, 2014, p. 2672–2680.

[21] ARJOVSKY, M.; LON, B. Towards principled methods for training generative adversarial networks. *NIPS 2016 Workshop on Adversarial Training. In review for ICLR*, 2017, vol. 2016.

[22] SALIMANS, T.; GOODFELLOW, I.; ZAREMBA, W.; CHEUNG, V.; RADFORD, A.; CHEN, X. Improved techniques for training gans. *Advances in neural information processing systems*, 2016, p. 2234–2242.

[23] YU, X.; FERNANDO, B.; HARTLEY, R.; PORIKLI, F. Super-resolving very low-resolution face images with supplementary attributes. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, p. 908–917.

[24] LI, D.; CHEN, D.; JIN, B.; SHI, L.; GOH, J.; NG, S. MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks. *International Conference on Artificial Neural Networks*, 2019, p. 703–716.

[25] ZHU, J.; PARK, T.; ISOLA, P.; EFROS, A. Unpaired image-to-image translation using cycle-consistent adversarial networks. *Proceedings of the IEEE international conference on computer vision*, 2017, p. 2223–2232.

[26] NIRKIN, Y.; KELLER, Y.; HASSNER, T. FSGAN: Subject Agnostic Face Swapping and Reenactment. *Proceedings of the IEEE International Conference on Computer Vision*, 2019, p. 7184–7193.

[27] THOMAS, C. *Deep learning based super resolution, without using a GAN* [online]. 2019, last actualization 1. 2. 2019 [cit. 20. 10. 2019]. Available: <`https://towardsdatascience.com/deep-learning-based-super-resolution-without-using-a-gan-11c9bb5b6cd5`>

[28] LIM, B.; SON, S.; KIM, H.; NAH, S.; MU LEE, K. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017. p. 136–144.

[29] DONG, C.; LOY, C. C.; HE, K.; TANG, X. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 2015, vol. 38, no. 2, p. 295–307.

[30] ZHANG, Y.; TIAN, Y.; KONG, Y.; ZHONG, B.; FU, Y. Residual dense network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. p. 2472–2481.

[31] KIM, K.; CHUN, SE Y. SREdgeNet: Edge Enhanced Single Image Super Resolution using Dense Edge Detection Network and Feature Merge Network. In *arXiv preprint arXiv:1812.07174*, 2018.

[32] KIM, D.; KIM, M.; KWON, G.; KIM, D. Progressive Face Super-Resolution via Attention to Facial Landmark. *arXiv preprint arXiv:1908.08239*, 2019.

[33] BULAT, A.; TZIMIROPOULOS, G. How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks). *Proceedings of the IEEE International Conference on Computer Vision*, 2017, p. 1021–1030.

[34] AGUSTSSON, E.; TIMOFTE, R. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017. p. 126–135.

[35] HUANG, J.; SINGH, A.; AHUJA, N. Single Image Super-Resolution From Transformed Self-Exemplars. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 06 2015.

[36] DAVID R. M.; CHARLESS C. F.; DORON T.; JITENDRA M. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, 2001, vol. 2, p. 416–423.

[37] LIU, Z.; LUO, P.; WANG, X.; TANG, X. Deep learning face attributes in the wild. *Proceedings of the IEEE international conference on computer vision*, 2015, p. 3730–3738.

[38] KHATTAB, M.; ZEKI, A.; ALWAN, A.; BADAWY, A. Regularization-based multi-frame super-resolution: A systematic review. *Journal of King Saud University-Computer and Information Sciences*, 2018.

[39] YUAN, Q.; ZHANG, L.; SHEN, H. Multiframe super-resolution employing a spatially weighted total variation model. *IEEE Transactions on circuits and systems for video technology*, 2011, vol. 22, no. 3, p. 379–392.

[40] SHI, F.; CHENG, J.; WANG, L.; YAP, P.; SHEN, D. LRTV: MR image super-resolution with low-rank and total variation regularizations. *IEEE transactions on medical imaging*, 2015, vol. 34, no. 12, p. 2459–2466.

[41] YANG, X.; LIU, T.; ZHOU, D. A multi-frame adaptive super-resolution method using double channel and regional pixel information. *Optik*, 2015, vol. 126, no. 24, p. 5850–5858.

[42] KIANI, K. A.; DRUMMOND, T. Solving robust regularization problems using iteratively re-weighted least squares. *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017, p. 483–492.

[43] KHATTAB, M.; ZEKI, A.; ALWAN, A.; BADAWY, A.; THOTA, L. S. Multi-Frame Super-Resolution: A Survey. *2018 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, 2018, p. 1–8.

[44] OLIVIER, R.; HANQIANG, C. Nearest neighbor value interpolation. *arXiv preprint arXiv:1211.1768*, 2012.

[45] ZHANG, X.; LIU, Y. A Computationally Efficient Super-Resolution Reconstruction Algorithm Based On The Hybird Interpolation. *JCP*, 2010, vol. 5, no. 6, p. 885–892.

[46] DE VOS, B.; BERENDSEN, F.; VIERGEVER, M.; SOKOOTI, H.; STARING, M.; IŠGUM, I. A deep learning framework for unsupervised affine and deformable image registration. *Medical image analysis*, 2019, vol. 52, p. 128–143.

[47] EPPENHOF, K.; LAFARGE, M.; MOESKOPS, P.; VETA, M.; PLUIM, J. Deformable image registration using convolutional neural networks. *Medical Imaging 2018: Image Processing*, 2018, vol. 10574, p. 105740S.

[48] USTINOVA, E; LEMPITSKY, V. Deep multi-frame face super-resolution. *arXiv preprint arXiv:1709.03196*, 2017.

[49] SCHOLLMEYER, A.; SCHNEEGANS, S.; BECK, S.; STEED, A.; FROEHLICH, B. Efficient hybrid image warping for high frame-rate stereoscopic rendering. *IEEE transactions on visualization and computer graphics*, 2017, vol. 23, no. 4, p. 1332–1341.

[50] SAJJADI, M.; VEMULAPALLI, R.; BROWN, M. Frame-recurrent video super-resolution. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, p. 6626–6634.

[51] CHU, M.; XIE, Y.; LEAL-TAIXÉ, L.; THUEREY, N. Temporally Coherent GANs for Video Super-Resolution (TecoGAN). *arXiv preprint arXiv:1811.09393*, 2018.

[52] WANG, X.; CHAN, K.; YU, K.; DONG, C.; CHANGE LOY, C. Edvr: Video restoration with enhanced deformable convolutional networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, p. 0–0.

[53] HITACHI, L. *Super-resolution technology to convert video of various resolutions to high-definition* [online]. 2008, [cit. 20.11.2019]. Available: <`http://www.hitachi.com/New/cnews/080924a.html`>

[54] YUE, L.; SHEN, H.; LI, J.; YUAN, Q.; ZHANG, H.; ZHANG, L. Image super-resolution: The techniques, applications, and future. *Signal Processing*, 2016, vol. 128, p. 389–408.

[55] NGUYEN, K.; FOOKES, C.; SRIDHARAN, S.; TISTARELLI, M.; NIXON, M. Super-resolution for biometrics: A comprehensive survey. *Pattern Recognition*, 2018, vol. 78, p. 23–42.

[56] GOHSHI, S. Real-time super resolution algorithm for security cameras. *2015 12th International Joint Conference on e-Business and Telecommunications (ICETE)*, 2015, vol. 5, p. 92–97.

[57] SHI, W.; CABALLERO, J.; HUSZÁR, F.; TOTZ, J.; AITKEN, A.; BISHOP, R.; RUECKERT, D.; WANG, Z. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, p. 1874–1883.

[58] AYYOUBZADEH, S. M.; WU, X. Adaptive Loss Function for Super Resolution Neural Networks Using Convex Optimization Techniques. *arXiv preprint arXiv:2001.07766*, 2020.

[59] BARE, B.; YAN, B.; MA, C.; LI, K. Real-time video super-resolution via motion convolution kernel estimation. *Neurocomputing*, 2019, vol. 367, p. 236–245.

[60] JOHNSON, J.; ALAHI, A.; FEI-FEI, L. Perceptual losses for real-time style transfer and super-resolution. *European conference on computer vision*, 2016, p. 694–711.

[61] ODENA, A.; DUMOULIN, V.; OLAH, C. Deconvolution and Checkerboard Artifacts. *Distill*, 2016, [cit. 20. 04. 2020]. Available: <`http://distill.pub/2016/deconv-checkerboard`>

[62] BINIELI, M. *Machine learning: an introduction to mean squared error and regression lines* [online]. 2019, [cit. 20.04.2020]. Available: <https://www.freecodecamp.org/news/machine-learning-mean-squared-error-regression-line-c7dde9a26b93/>

[63] BHATTACHARYA, A.; CHATTERJEE, T. An Estimation Method of Measuring Image Quality for Compressed Images of Human Face. *arXiv preprint arXiv:1402.1331*, 2014.

[64] NARVEKAR, N.; KARAM, L. A no-reference perceptual image sharpness metric based on a cumulative probability of blur detection. *2009 International Workshop on Quality of Multimedia Experience*, 2009, p. 87–91.

List of symbols, physical constants and abbreviations

CNN	Convolution Neural Network
CPU	Central processing unit
EDSR	Enhanced Deep Super Resolution Network
FCN	Fully Convolutional Network
GAN	Generative Adversarial Network
GPU	Graphics Processing Unit
IRLS	Iteratively Re-weighted Least Squares
LR	Low Resolution
MISR	Multi-frame Image Super Resolution
MRI	Magnetic Resonance Imaging
MSE	Mean Square Error
PET	Positron Emission Tomography
PSNR	Peak Signal-to-Noise Ratio
RDN	Residual Dense Network
ReLU	Rectified Linear Unit
SISR	Single Image Super Resolution
SR	Super Resolution
SRCNN	Super-Resolution Convolutional Neural Network
SRGAN	Super Resolution Generative Adversarial Network
SSD	SingleShot MultiBox Detector
SSIM	Structural Similarity
TV	Total Variation

List of appendices

A The contents of the attachment

```
/
├── models ..... Implemented models
│   ├── unet_geu
│   │   ├── __init__.py
│   │   └── model.py ..... Script with model "U-Net + GEU"
│   ├── unet_geu_2
│   │   ├── __init__.py
│   │   ├── model.py ..... Script with model "U-Net + GEU 2"
│   │   └── __init__.py
│   ├── unet_geu_3
│   │   ├── __init__.py
│   │   └── model.py ..... Script with model "U-Net + GEU 3"
│   ├── unet_resblocks
│   │   ├── __init__.py
│   │   └── model.py ..... Script with model "U-Net + ResBlocks"
│   ├── data.py ..... Script for preparation data
│   ├── loss_functions.py ..... Script with loss functions
│   ├── multi_test.py ..... Script for testing models
│   └── train_multi.py ..... Script for training models
├── CompareImages.py ..... Script for computation metrics
├── filters.py ..... Script to apply the filters to images
└── readme.txt
```