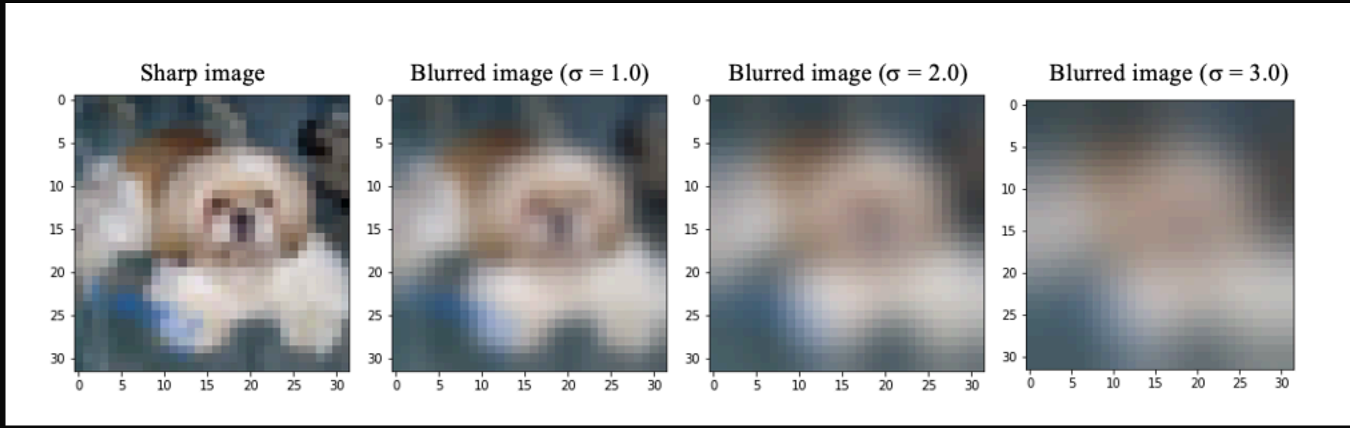
# Deep Image Deblurring

Eleonora Mancini, Marcello Simonati

# Contents

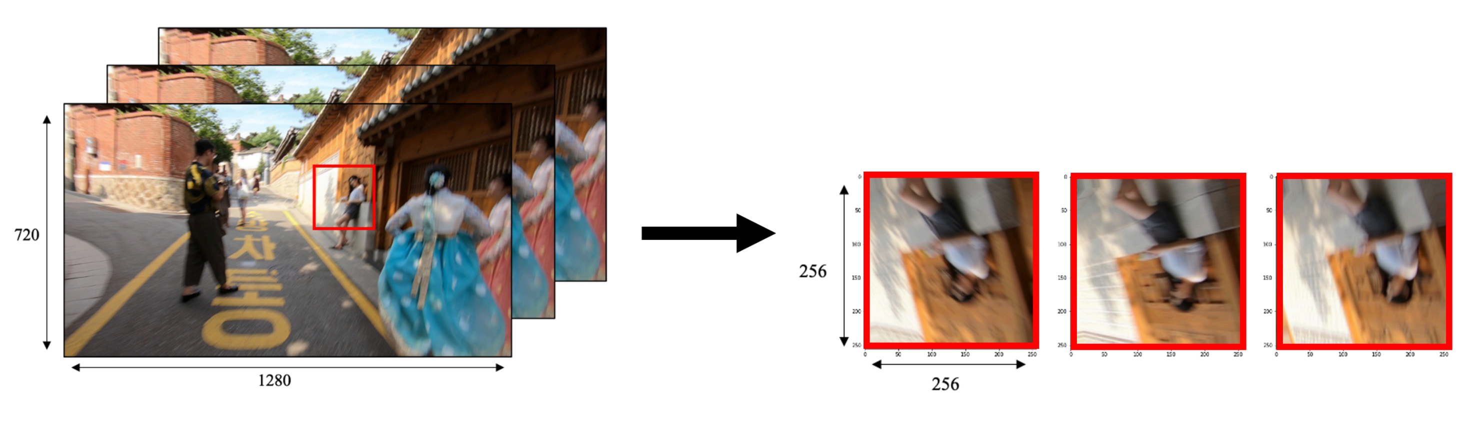
# Datasets
## CIFAR-10

- 60 000 images of resolution 32x32.
- **Preprocessing**
  - Every image was smoothed with a Gaussian Kernel $0 \leq rand(\sigma) \leq 3$
  - No data augmentation

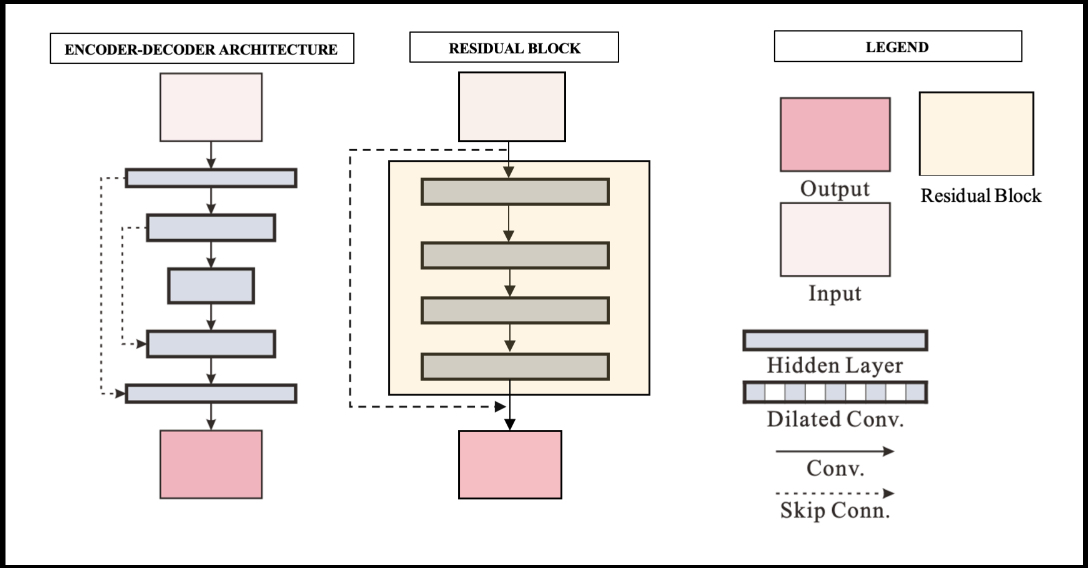


# Datasets
## REDS

- 300 videos with 100 images of resolution 720 × 1280 each
- **Preprocessing**
  - Random image **crops** of 256 x 256
  - Vary **brightness** by small amount for every image crop
  - **Flip** and rotate **images** at a 50 % rate each
  - Concatenation of previous and next frame, shifted up to 2 pixels to the opposite directions respectively



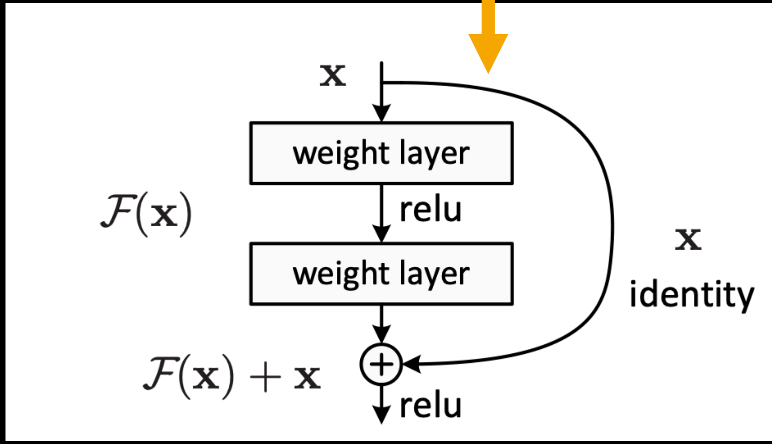
# Basic ideas behind our architectures

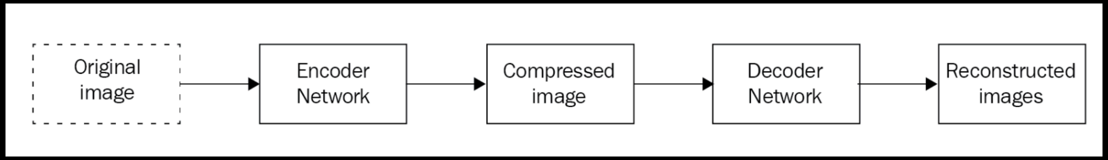


# Basic ideas behind our architectures

## Overview

- Skip Connections
  - Addition / Concatenation
  - Short / Long
- Residual Blocks
- Convolutional Autoencoders



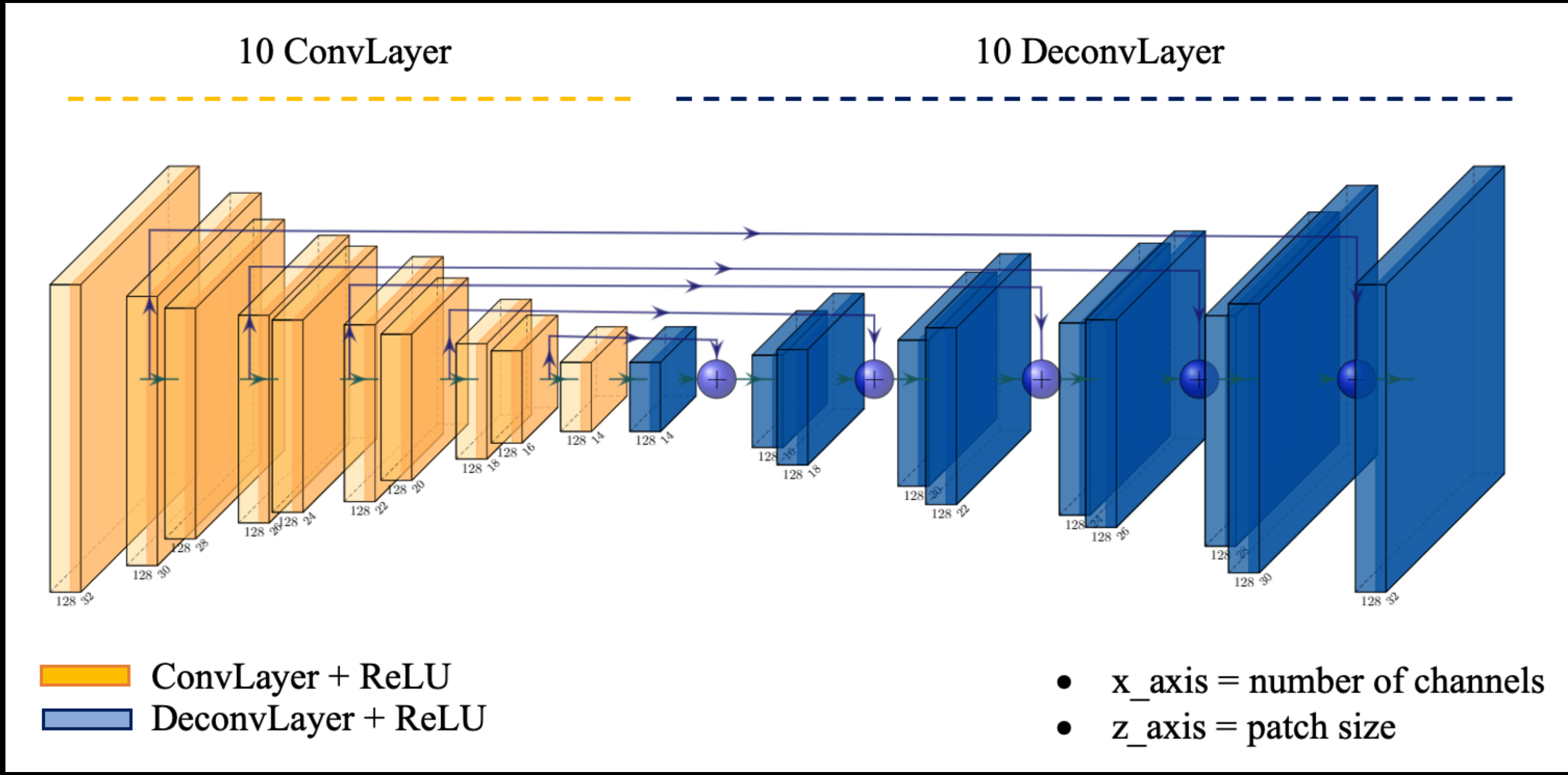



# ARCHITECTURES

## 1. CARLO_NET

TESTED ONLY ON CIFAR-10

- **Convolutional Autoencoder** with the use of symmetric **skip connections.**
- **Skip connections**:
  - **faster training** + **better result**
  - **every 2 layers** from convolutional features maps to their mirrored deconvolutional feature maps
- **Activation:** ReLU
- **Input and Output:** input size can be arbitrarily, input and output have the same size
- **Kernel Size:** 3x3
- **Padding:** Valid (i.e. no padding)
- **Stride:** 1 (i.e. no stride)

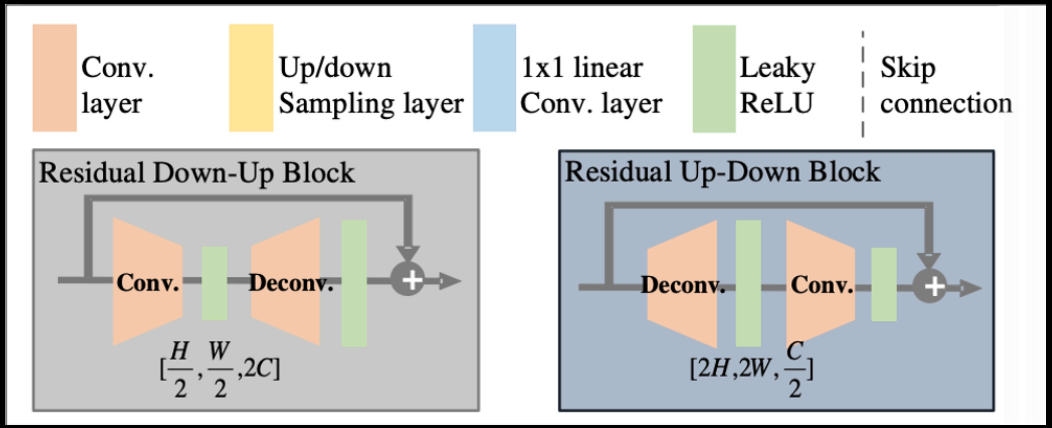


# ARCHITECTURES

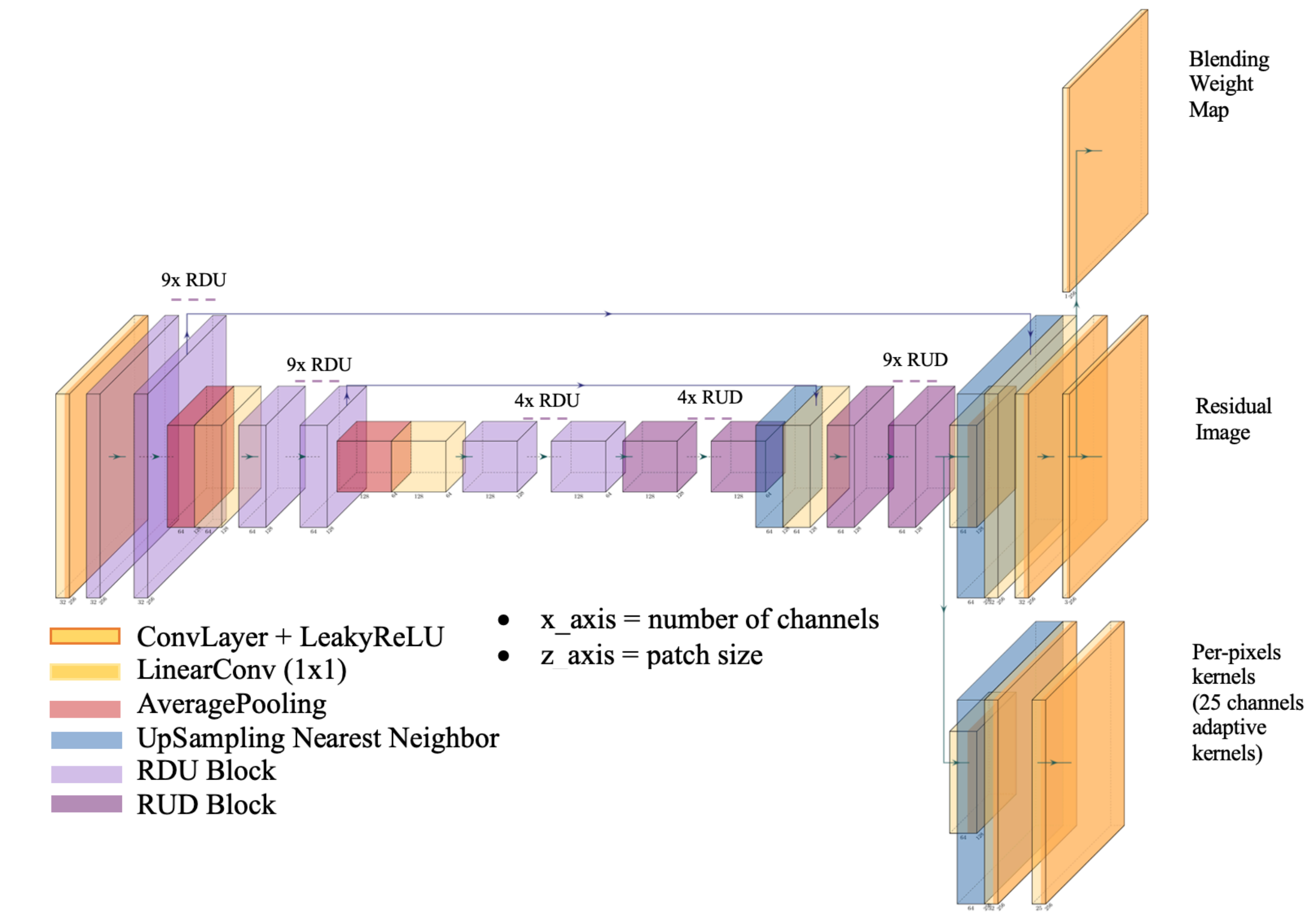
## 2. KAIST_NET

**TESTED BOTH ON CIFAR-10 AND REDS**

- **Convolutional Autoencoder** based on the idea of the U-Net architecture, where the covolution layers in the encoder and decoder parts are replaced with residual down-up and residual up-down blocks respectively.
- **Activation:** Leaky ReLU
- **Res Down-Up Block**:
  - Convolution 5x5, stride 2, padding same
  - Leaky ReLU
  - Deconvolution 4x4, stride 2, padding same
- **Res Up-Down Block**:
  - Deconvolution 4x4, stride 2, padding same
  - Leaky ReLU
  - Convolution 5x5 , stride 2, padding same



# ARCHITECTURES
## 2. KAIST_NET - STRUCTURE



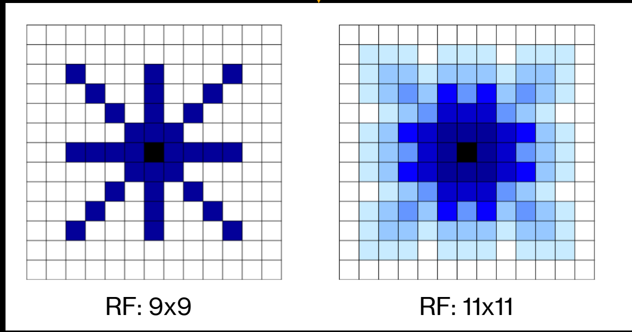

- **Output**
  - $L = w \cdot B \times K_d + (1 - w) \cdot R$
  - **Sum of:**
    - **Weighted residual RGB image *R***
    - **Adaptive convolution**
      - **25 channels adaptive kernels *K***
      - **Blurred image *B***
      - **Blending weight map *w***

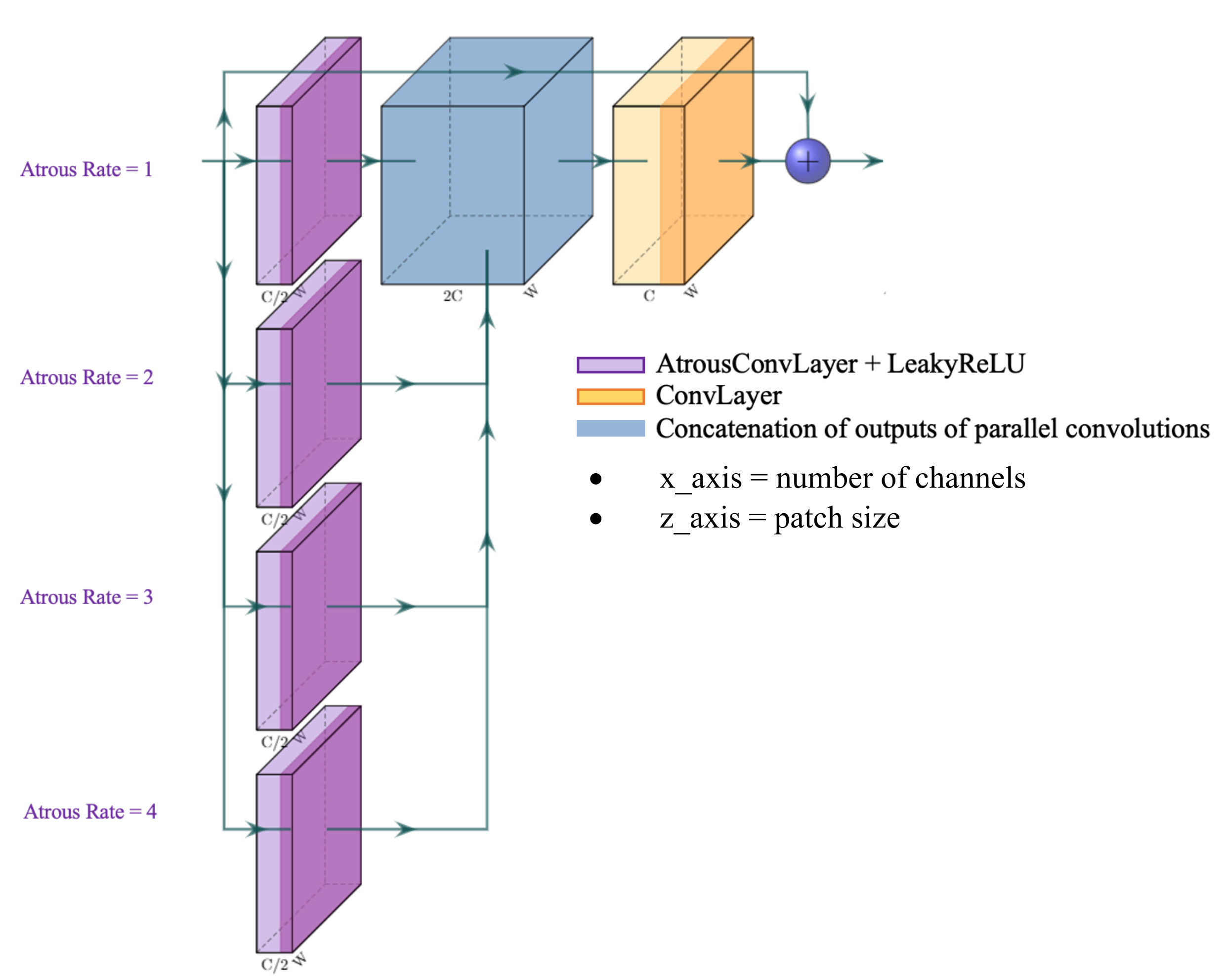
# ARCHITECTURES

## 3. ATROUS_NET - Official network of our work

TESTED BOTH ON CIFAR-10 AND REDS

- **Convolutional network** based on the use of novel **Residual Blocks** composed **of Atrous (or dilated) convolutions**
- Every **Convolution** in this block has:
    - **Kernel Size:** 3x3
    - **Stride:** 1 (i.e. no stride)
    - **Padding**: Same
- **Receptive field** of each Atrous Block : 11 x 11



# ARCHITECTURES
# 3. ATROUS_NET : WHY USE ATROUS CONVOLUTIONS?

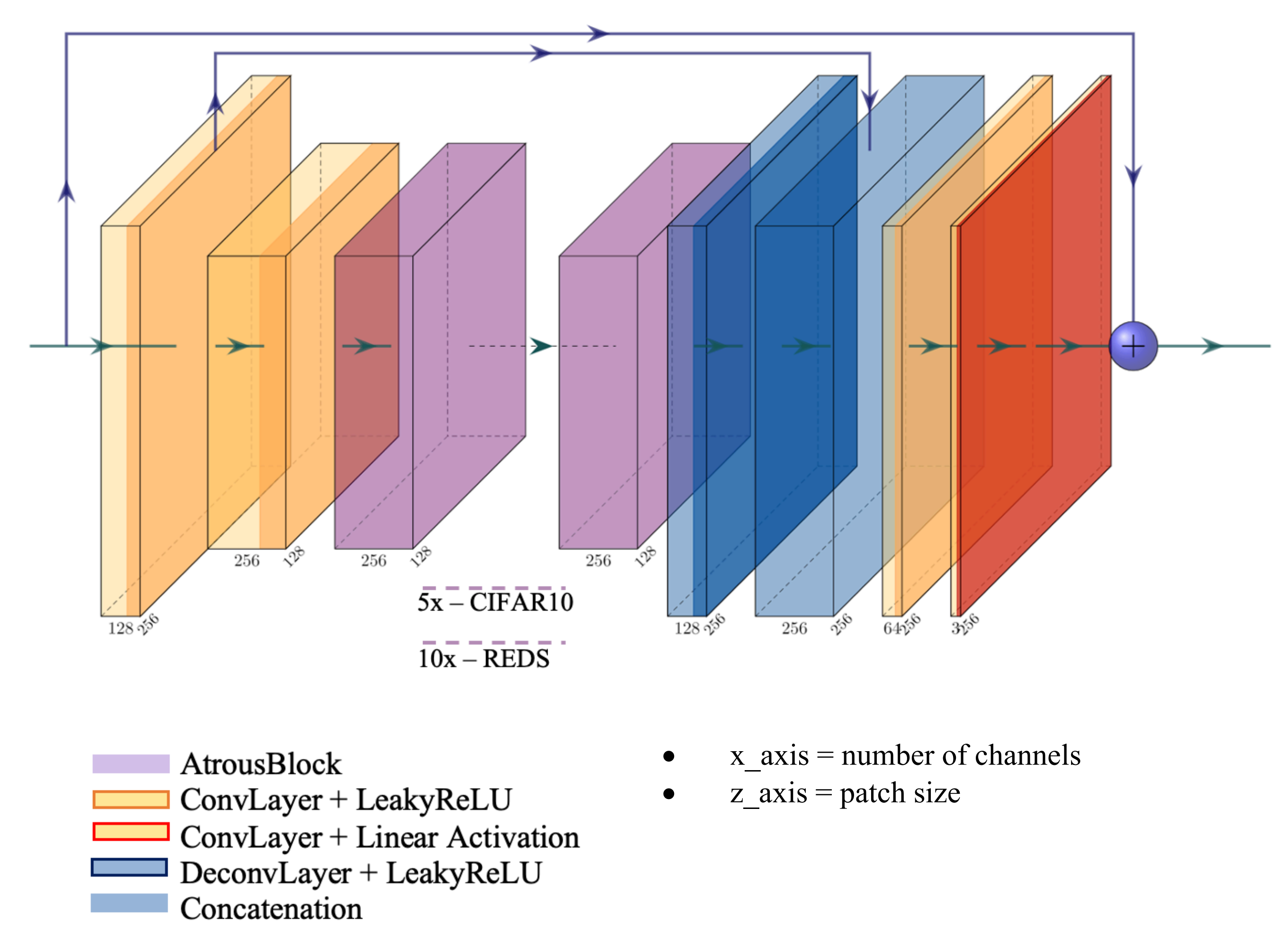
- Simulate a **multiscale approach without reducing spatial resolution**
    - by using 3x3 atrous convolutions at different dilation rates we can achieve a very wide receptive field at a much lower computational cost (w.r.t. a comparable standard convolution), without losing too much spatial resolution.

# ARCHITECTURES

## 3. ATROUS_NET - STRUCTURE

**TESTED BOTH ON CIFAR-10 AND REDS**

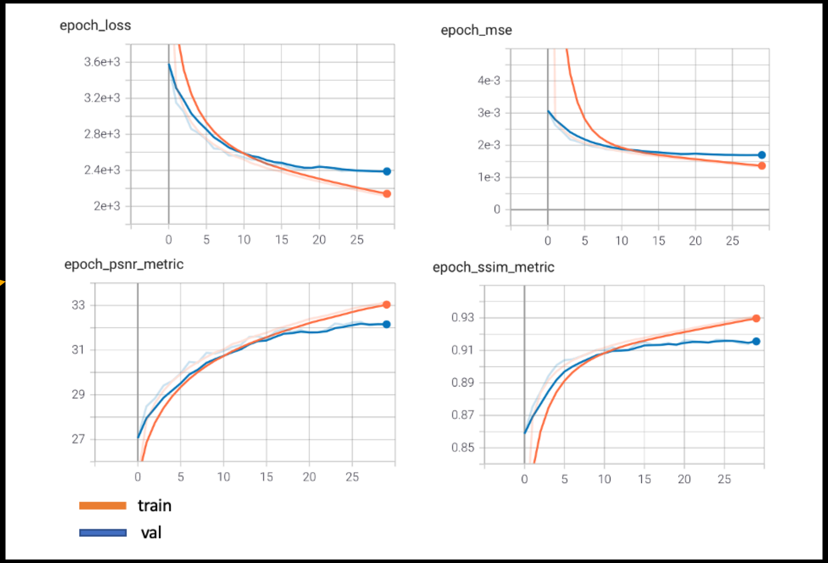
- <u>**Inner structure**</u>
  - **Convolution 9x9, padding same**
    - ➢ low-level features extraction
  - **Convolution 3x3, stride 2, padding same**
    - ➢ down-sampling
  - Series **of Atrous Blocks**:
    - **CIFAR-10**: 5 Residual Atrous Blocks
    - **REDS**: 10 Residual Atrous Blocks
  - **Deconvolution 3x3** and **2 Convolutions 3x3, stride 2, padding same**
    - ➢ up-sampling
- <u>**Output**</u>: sum of blurred input image and residual image
- <u>**Weight Initialization:**</u> He Initializer



# TRAINING

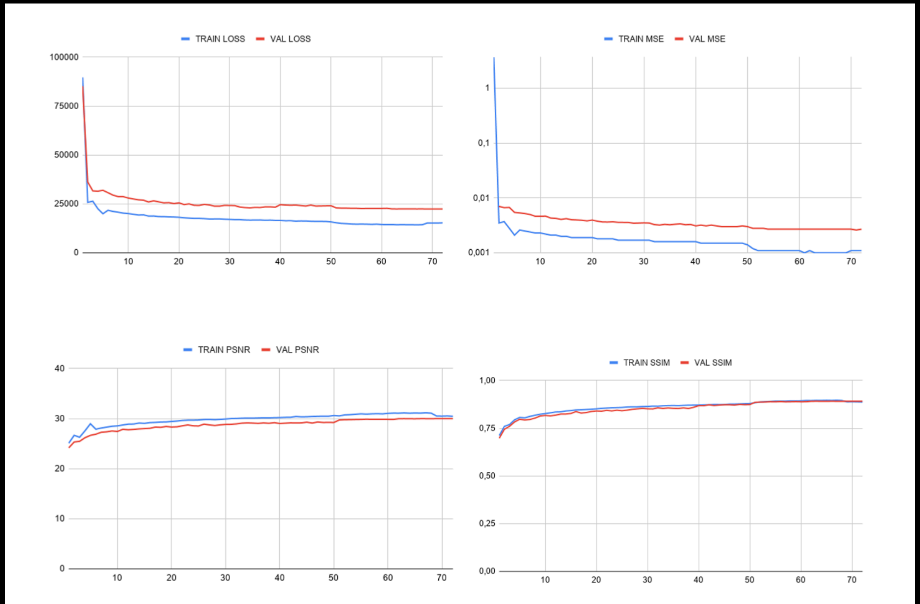
## CIFAR-10

| | BATCH SIZE | ITERATIONS | OPTIMIZER | LOSS |
|---|---|---|---|---|
| **CARLO_NET** | 32 | 40 | Adam | MSE/L2 |
| **KAIST_NET** | 32 | 30 | Adam | LAD/L1 |
| **ATROUS_NET** | **32** | **29** | **Adam** | **LAD/L1** |



# TRAINING
## REDS

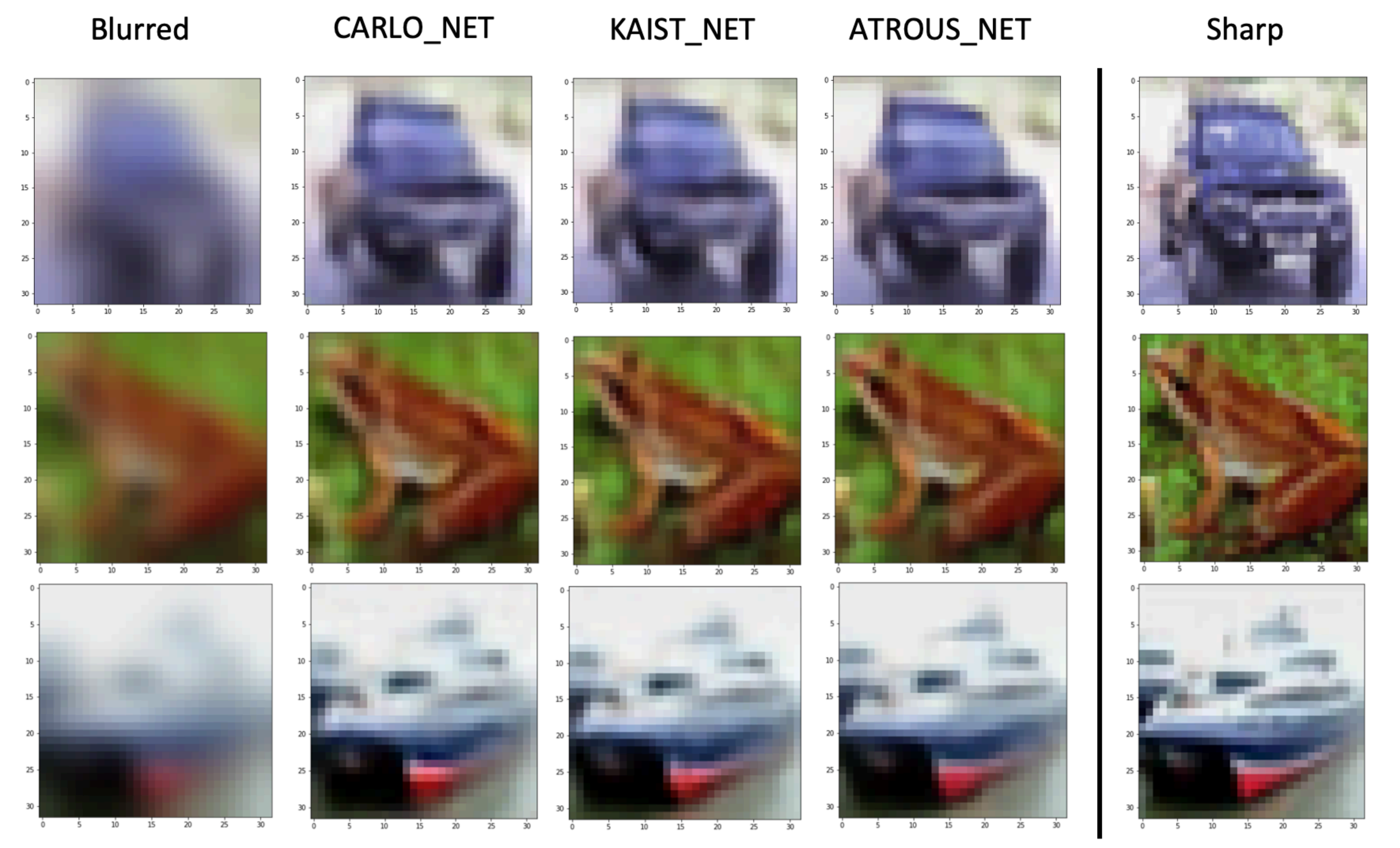
| | BATCH SIZE | ITERATIONS | OPTIMIZER | LOSS |
|---|---|---|---|---|
| **KAIST_NET** | 4 | 55 | Adam | LAD/L1 |
| **ATROUS_NET** | **4** | **71** | **Adam** | **LAD/L1** |



# EXPERIMENTAL RESULTS

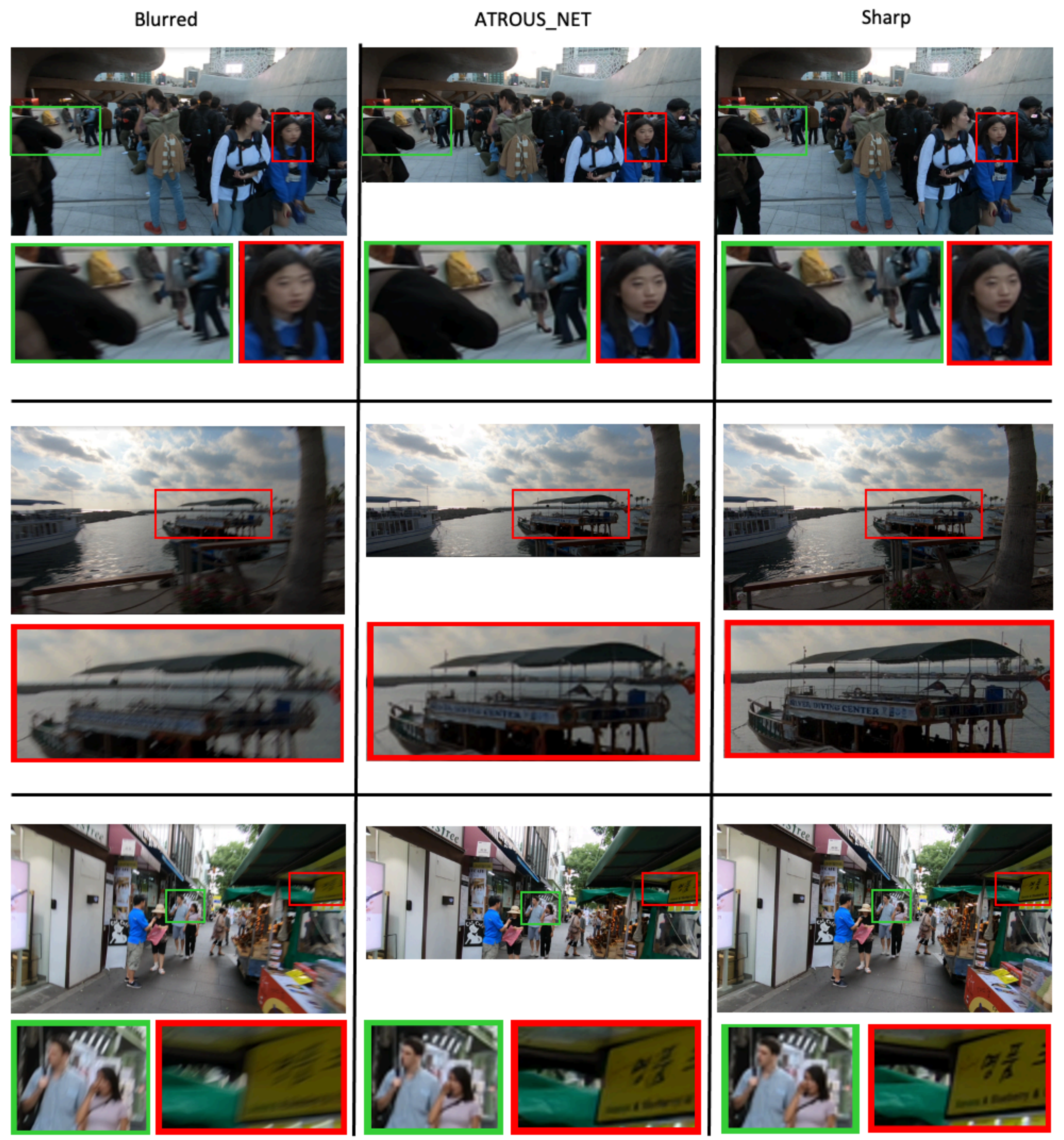
## CIFAR-10

| | CARLO_NET | KAIST_NET | ATROUS_NET | Baseline |
|---|---|---|---|---|
| **SSIM** | 0.905 | 0.9012 | **0.9237** | 0.7127 |
| **PSNR** | 29.34 | 31.53 | **33.61** | 24.64 |
| **MSE [$10^{-3}$]** | 2.11 | 1.77 | **1.50** | 6.34 |


Blurred
CARLO_NET
KAIST_NET
ATROUS_NET
Sharp

# EXPERIMENTAL RESULTS
## REDS

| | KAIST_NET | UniA Team | DeblurGAN – v2 | ATROUS_NET | Baseline |
|---|---|---|---|---|---|
| **SSIM** | 0.806 | 0.9412 | 0.8059 | **0.9009** | 0.7617 |
| **PSNR** | 28.70 | 34.44 | 28.92 | **32.42** | 27.048 |
| **MSE [$10^{-3}$]** | 2.42 | ---------------- | ---------------- | **1.05** | 3.88 |


Blurred
ATROUS_NET
Sharp

# ROBUSTNESS BETA-TEST
## ATROUS_NET ON GOPRO

- **Beta** because this test has been carried out without training ATROUS_NET on GoPro
- Although the network is not trained on GoPro, we can say that **it still manages to slightly remove the noise present in the images**