A Hybrid Approach to EDVR and YOLOv8 for Super-Resolution

By

Abul Monsur Mannan

Submitted to the
Dept. of Computer and Software Engineering
Technological University of the Shannon

In partial fulfilment of the requirements for the degree of

MSc in Software Design with Artificial Intelligence
Technological University of the Shannon
August 2023

Supervised by: **Dr. Yuhang Ye**

Declaration of Authorship

I, Abul Monsur Mannan, declare that this thesis titled, '**A Hybrid Approach to EDVR and YOLOv8 for Super-Resolution**' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this Institute.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.

Signed: Amonsur
(Abul Monsur Mannan)

Date: 03-09-2023

Acknowledgements

I would like to thank my supervisor Dr Yuhang Ye for providing invaluable support and help in developing and writing this research work. I am grateful to my supervisor for generously giving me directions and suggestions for my paper revisions and the direction of the work and paper submissions. I must also offer my thanks and appreciation to Dr Enda Fallon for all of his support and help.

Abstract

Advances in super-resolution (SR) and object detection have often been used largely in parallel within the domain of computer vision. While ‘Enhanced Deformable Video Restoration’ (EDVR) excels in generating high-resolution images, models like YOLOv8 are optimized for detecting and segmenting objects within those images. This research seeks to synergize these specialized capabilities into a unified framework while adopting a ‘qualitative methodology’ approach. The study fuses the aligned feature maps produced by EDVR with YOLOv8-seg's contextual layers on a standard Set5 dataset utilizing a PyTorch-based implementation. The hypothesis posits that the fusion of features from both models will result in high-quality super-resolution outputs, augmented with precise object detection and segmentation. The proposed model has applications across various domains, including surveillance, medical imaging, and autonomous vehicles.

Table of Contents

TABLE OF CONTENTS I
LIST OF FIGURES IV
LIST OF TABLES V
ACRONYMS VI

CHAPTER 1: INTRODUCTION 1
1.1 Background 1
1.2 Research Gap 1
1.3 Research Objective 1
1.4 Methodology 1
1.5 Hypothesis 2
1.6 Structure of the Thesis 2
1.7 Significance of the Study 2
CHAPTER 2: LITERATURE REVIEW 3
2.1 Deep Learning Models for Super Resolution 4
2.2 Video Super Resolution (VSR) 6
2.3 Techniques/methods used in SR tasks 8
2.3.1. Spatial & Temporal information in Super Resolution problems: 8
2.3.2. Techniques to extracting spatial and temporal information 9
2.4 Datasets for Super Resolution 11
2.5 Comparison of Super Resolution Methods 13
2.6 About the base models for our research 14
2.6.1 An overview on EDVR model: 14
2.6.1.1 EDVR: Strengths and Weaknesses 16
2.6.2 An overview on YOLOv8-seg model: 17
CHAPTER 3: PROBLEM STATEMENT 18
3.1 Challenges and Limitations of Traditional Approaches 19
CHAPTER 4: RESEARCH OBJECTIVES 20
CHAPTER 5: RESEARCH METHODOLOGY 21
5.1 Overview 21
5.2 Methodology breakdown 25
5.2.1 Evaluation of YOLOv8 and EDVR 26

5.2.1.1 Architecture of EDVR:.... 26
5.2.1.2 Architecture of YOLOv8-seg:.... 28
5.2.1.2.1 The detection blocks: There are 3 detection blocks in the model. The primary reasons for having multiple detection blocks in YOLOv8 are:.... 31
5.2.1.2.2 The 'concat' blocks:.... 32
5.2.2 An analysis and uses of feature maps:.... 33
5.2.2.1 Initial Feature Extraction:.... 34
5.2.2.2 Temporal and Spatial Attention (TSA) Fusion Module:.... 34
5.2.2.3 Reconstruction of Module:.... 35
5.2.2.4 Deblurring Module:.... 35
5.2.3 Output:.... 36
5.2.4 Summary:.... 36
5.3 The foundation of EDVR is DCN.... 37
5.4 Factors that influence the performance and quality of image/video super-resolution models like EDVR.... 38
5.4.1 Data and Preprocessing.... 38
5.4.2 Loss Functions.... 38
5.4.3 Training Strategies.... 39
5.4.4 Computational Resources.... 39
5.4.5 Evaluation Metrics.... 39
5.4.6 External Factors.... 40
5.5 Implementation background.... 40
5.5.1 SEEM:.... 40
5.5.2 Improved EDVR:.... 42
5.6 Programming tools and frameworks.... 44
5.7 Conclusion.... 46
CHAPTER 6: THE PROPOSED MODEL.... 46
6.1 OUR MODULE HYPOTHESIS:.... 47
6.1.1 A graphical view of the proposed model.... 48
CHAPTER 7: EXPERIMENTS, RESULT AND DISCUSSION.... 50
7.1 YOLOv8-seg uses:.... 50
7.2 EDVR uses:.... 51
7.3 Developing our proposed module:.... 53
7.3.1 Feature extraction from YOLOv8:.... 53
7.3.2 Use of EDVR model on extracted feature maps from YOLOv8:.... 55
7.3.3 The final model structure:.... 57
7.3.4 Reason for the fusion models:.... 57
7.4 Results and discussion:.... 58

7.5 Implementation problems..... 60
CHAPTER 8: CONCLUSION..... 60
8.1 Methodological Foundations:..... 60
8.2 Proposed Model:..... 61
8.3 Experimental Outcomes:..... 61
8.4 Limitations and Future Work:..... 61
8.5 Implications:..... 61
CHAPTER 9: REFERENCES..... 62

List of Figures

FIGURE 1: A TYPICAL CNN ARCHITECTURE ---------- 5
FIGURE 2: A GRAPHICAL REPRESENTATION OF SPATIAL & TEMPORAL INFORMATION ---------- 9
FIGURE 3: EDVR: VIDEO RESTORATION WITH ENHANCED DEFORMABLE ---------- 28
FIGURE 4: YOLOV8 ARCHITECTURE IN DETAIL ---------- 30
FIGURE 5: PCD(LEFT) AND TSA (RIGHT) OF *EDVR* ---------- 35
FIGURE 6: SEEM MODEL'S ARCHITECTURE ---------- 41
FIGURE 7: A GRAPHICAL REPRESENTATION OF THE PROPOSED MODEL, TOP STRUCTURE IS THE EDVR MODEL AND BOTTOM ONE THE YOLOV8 MODEL ---------- 49
FIGURE 8: TRAINING WITH YOLOV8-SEG.YML ---------- 51
FIGURE 9: FEATURE MASKS FROM YOLOV8-SEG ---------- 51
FIGURE 10: EVDR INSTALLATION ---------- 52
FIGURE 11: RESULTS FROM THE EDVR PROJECT ---------- 53
FIGURE 12: FEATURE EXTRACTION FROM YOLOV8-SEG MODEL ---------- 54
FIGURE 13: SAVING FEATURE MAPS FROM YOLOV8-SEG MODEL ---------- 54
FIGURE 14, EDVR_ARCH CALL ---------- 55
FIGURE 15, FUSION WITH THE NEIGHBOURING FEATURES OF EDVR MODEL ---------- 55
FIGURE 16, THE FINAL MODEL ARCHITECTURE ---------- 57
FIGURE 17, COMPARING OUTPUT WITH THE GROUND TRUTH CALENDAR IMAGE FRAME FROM SET5 DATASET ---------- 58
FIGURE 18, COMPARING OUTPUT WITH THE BLURRED LR CALENDAR IMAGE FRAME FROM SET5 DATASET ---------- 59

List of Tables

TABLE 1:LIST OF DATASETS FOUND IN MANY RESEARCH PAPERS 13
TABLE 2:QUANTITATIVE RESULTS OF EDVR 15
TABLE 3: YOLOV8 MODEL'S PROMISING RESULTS 17
TABLE 4:YOLOV8'S DIFFERENT VERSIONS 51

Acronyms

LR	Low-Resolution
HR	High-Resolution
CNN	Convolutional Neural Network
SR	Super-Resolution
HD	Higher Definition
SRCNN	Super Resolution Convolutional Neural Network
DRCN	Deeply-Recursive Convolutional Network
EDSR	Enhanced Deep Super Resolution
GAN	Generative Adversarial Networks
SRGAN	Super Resolution Generative Adversarial Network
EDVR	Enhanced Deformable Convolutional Networks
EDVR	Enhanced Deep Video Restoration
VSR	Video Super Resolution
RNN	Recurrent Neural Network
DUF	Deep Video Upsampling
TDAN	Temporal Difference Adversarial Network
LSTM	Long Short-Term Memory
TGANs	Temporal Generative Adversarial Networks
SISR	Single Image Super Resolution
VSRGAN	Video Super-Resolution Generative Adversarial Network
ISR	Image Super Resolution
SRCNN	Super-Resolution Convolutional Neural Network
VDSR	Very Deep Super Resolution
PSNR	Peak Signal Noise Ratio
SSIM	Structural Similarity Index
DSR	Deep Super Resolution
VQM	Video Quality Metric
SSIM VQM	Structural Similarity-based Video Quality Metric
TSA	Temporal and Spatial Attention
SEEM	SAM-guided Ed refinement Module
FLOPs	Floating Point Operations
NLP	Natural Language Processing
ROI	Region-of-Interest

CSP	Cross-Stage Partial
DCN	Deformable Convolutional Network
SAM	Segment Anything Model
YOLOv8	You Only Look Once Version 8
VSR	Video Super-Resolution
ISR	Image Super-Resolution
IDE	Integrated Development Environment
PCD	Pyramid, Cascading and Deformable
TSA	Temporal Spatial Attention
CPU	Central Processing Unit
GPU	Graphics Processing Unit
ReLU	Rectified Linear Unit)

Chapter 1: Introduction

1.1 Background

In the field of computer vision, Super-Resolution (SR) stands as a pivotal technique aimed at enhancing the quality of images and video frames. Existing models such as Enhanced Deformable Video Restoration (EDVR) [8] have demonstrated significant progress in achieving Super Resolution. However, the integration of object detection and segmentation capabilities into SR remains a topic of intense research. To this end, models like YOLOv8-seg [4] provide noteworthy image segmentation capabilities, particularly when tasked with understanding the contextual information within images.

1.2 Research Gap

While EDVR specializes in video super-resolution, and YOLOv8-seg excels in object detection and instance segmentation, there is a lack of methodologies that combine the strengths of both to improve the super-resolution quality of images and video frames in a holistic manner.

1.3 Research Objective

The primary objective of this research is to propose a novel model that amalgamates the feature maps of YOLOv8-seg with the aligned features of EDVR. By doing so, this research aims to achieve high-quality super-resolution outputs while preserving precise object detection and segmentation.

1.4 Methodology

This study adopts a qualitative research approach, using Python-based PyTorch as the programming language and implementing the model through Jupyter Notebook and VSCode.

The initial development is conducted on a local CPU, followed by GPU acceleration via Google Colab. Datasets such as VID4, Set5, Set14, and possibly from the REDS or Set dataset, will be employed for empirical validation.

1.5 Hypothesis

The research hypothesizes that the fusion of feature maps from YOLOv8-seg and EDVR will produce high-quality image and video frames that not only exhibit enhanced perceptual quality but also include accurate object detection masks. This hypothesis inspired by the work like Fast-SAM (Zhao et al., 2023) [50], SEEM (Lu et al., 2023) [40] and Improved EDVR by Lu et al., (2023) [38].

1.6 Structure of the Thesis

The thesis is structured as follows:

Chapter 2 provides a literature review on existing models and techniques.

Chapter 3 outlines the problem statement.

Chapter 4 elucidates on the research objective.

Chapter 5 elaborates on the research methodology.

Chapter 6 introduces the proposed model and its underlying hypothesis.

Chapter 7 outlines the experimental setup, results, and discussions.

Chapter 8 concludes the study and offers suggestions for future research.

1.7 Significance of the Study

This research has the potential to significantly contribute to the field of computer vision, particularly in tasks requiring super-resolution along with object detection and segmentation. It could pave the way for applications in various domains such as surveillance, medical imaging, and autonomous driving systems.

Chapter 2: Literature Review

Numerous researchers have put forth learning models to tackle the issue of super-resolution, each, with its unique advantages and limitations. One of the methods employed was interpolation, a widely used and straightforward technique, for enhancing image resolution (Kaur et al., 2021). To overcome these limitations, researchers have come up with techniques, like sparse coding and dictionary learning. According to Ayas & Ekinci (2020)[5], these methods aim to find a representation of the low-resolution image and understand how it maps to the high-resolution space. However, these techniques are computationally demanding and at times do not work efficiently with datasets. Another popular approach involves using patch-based methods, such as the self-model and non-local means. These methods split the LR (lower resolution) image into overlapping patches and there is a need to investigate more about how these patches relate to their HR counterparts (Chen, 2020) [14]. By exploring more about higher-resolution image improvement, the advancements allow for the reconstruction of high-resolution details based on the information gathered from low-resolution patches. Patch-based methods have proven effective in restoring textures and fine details. They can sometimes introduce blocking artefacts that need careful parameter adjustments. Hence, the literature has explicated some popular techniques for super-resolution along with the explication that LR imaging in HD (higher definition) models have improved a lot and there is a need for HR to be explored and improved by the researchers and engineers for improved resolution.

2.1 Deep Learning Models for Super Resolution

In the last few years of development of deep learning models, specifically CNNs have shown impressive achievements in various computer vision tasks where one of the eminent task is super-resolution. According to a study conducted by Wang et al. (2020) [44], these models can effectively capture nonlinear relationships between Low-Resolution (LR) and High-Resolution (HR) images allowing them to restore fine details and produce visually pleasing outcomes. One of the learning models designed for super-resolution is called Super Resolution Convolutional Neural Network (SRCNN) which was redesigned by Dong *et al.* (2016) [15]. SRCNN comprises layers of operations and nonlinear activation functions followed by a deconvolution layer that magnifies the image. At the time of its introduction, SRCNN delivered cutting-edge performance where the organization has set the foundation towards advancement in the domain of super-resolution. According to Wang et al. (2020), several variants and enhancements of SRCNN have been anticipated and the advancements of SRCNN includes the Very Deep Convolution Network. Moreover, according to Kim et al. (2016) [24], Deeply-Recursive Convolutional Network (DRCN), and Enhanced Deep Super Resolution (EDSR) from Lim *et al.* (2017) [59] are two other techniques that can be used for super-resolution. The DRCN is a type of convolutional neural network (CNN) architecture, which is created for image super-resolution tasks and is characterized by its deep recursion. This means that it involves multiple recursive stages where the input image is progressively refined and, in each recursion, low-resolution images are passed through convolutional layers to enhance their spatial resolution.

Whereas, EDSR is a type of deep learning technique for single-image super-resolution, which focuses on the use of a very deep architecture to capture fine details in high-resolution images. The EDSR architecture typically involves several blocks of convolutional layers that

employs residual connections for mitigating the vanishing gradient problem and facilitating training. However, these prototypes are typically used for deeper and wider network architectures, and therefore, it results in improved performance and higher-quality super-resolved images and videos.

Classical methods that are used for super resolution include neighbour, bilinear, and bicubic interpolation techniques are commonly used in traditional image processing (Singh & Singh, 2019) [34]. However, their effectiveness is limited when it comes to handling textures or high-frequency details. On the other hand, deep learning-based SR models have made progress compared to these classical approaches. Among the learning models used for SR, Convolutional Neural Networks (CNNs) have gained popularity due to their ability to learn the relationship between resolution and high-resolution images (Maggiori et al., 2017) [30].

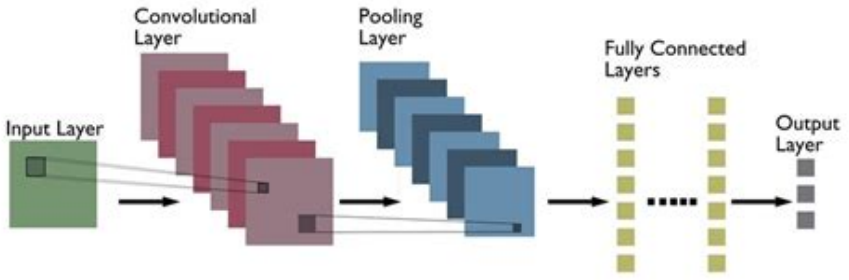

Figure 1: A typical CNN architecture. Source: [34]

One example of such a model is Super Resolution Convolutional Neural Network (SRCNN) which utilizes three layers of CNNs to learn this mapping function. SRCNN has proven to outperform SR algorithms like interpolation.

Moreover, with the use of CNNs Generative, Adversarial Networks (GANs) have also demonstrated the potential of enhancement of images in the field of SR. GAN-based models employ a network that generates top-notch images and another network that distinguishes between resolution and generated images (Liu et al., 2022) [27]. Another model, SRGAN by

(Ledig *et al.*, 2016) [9] has demonstrated outcomes by generating high-quality images that are challenging to differentiate from their high-resolution counterparts. Video SR presents a task compared to a single image as SR is not only a source of producing images, but it increases the quality-frames from inferior video sequences to enhanced versions. To accomplish SR in the domain of video, there is a need for the utilization of information, for generating frames. Recently, deep learning models tailored for video SR have emerged, one example being Enhanced Deep Video Super Resolution (EDVR) by Wang et al. (2019) [7]. EDVR integrates spatial information to enhance the quality of video frames and has outperformed models exhibiting promising outcomes in producing high-quality videos.

In brief, deep learning-based models have showcased progress in the field of SR surpassing approaches. CNN-based models like SRCNN and VDSR [69] have demonstrated state-of-the-art performance in single-image SR while GAN-based models like SRGAN can be evidently used for SR in the future. Additionally, advanced deep learning models specifically designed for video SR, such, as EDVR have been proposed with encouraging results. Ongoing research is expected to lead to advancements and impactful deep-learning models for SR.

2.2 Video Super Resolution (VSR)

A commendable survey on VSR models by Liu et al. (2022) states that the techniques which are used for the creation of high-resolution video from low resolution, are based on deep neural networks and therefore, they have made great progress. The study was conducted using a systematic review where 37 state-of-the-art VSR techniques were investigated. The result of the study has indicated that information or data, which are contained in the video frame are very significant for video super-resolution as they improve the picture quality and make it look

clearer. Per the findings of Liu et al. (2022), when it comes to improving image quality most of the research focuses on Single Image Super Resolution (SISR). However, Video Super Resolution (VSR) takes image resolution a step further by enhancing the quality of the video by enhancing the information between frames. VSR models commonly utilize networks (RNNs) or 3D Convolutional Neural Networks (CNNs) to capture the sequential patterns and generate high-quality frames. Some known VSR models include DUF (Deep Video Upsampling) [26]and EDVR (Enhanced Deep Video Restoration).

From the study conducted by Geng et al. (2022) [16], it is established that video super-resolution methods that have been used, make use of spatial information most of the times produces high-resolution video frames. These approaches leverage the capabilities of networks (CNNs) and recurrent neural networks (RNNs) to achieve higher-quality results. In their paper (Geng et al., 2022)[16], they used modern transformer-based architecture to gather special and temporal information about the video frames and they saw success in their approach.

Spatial information refers to the details and structures present within an image and, in the video, encompasses the visual features and patterns that make up each individual frame (Zeng et al., 2019). When working with video sequences, there is often a strong correlation between consecutive frames due to the temporal coherence of the content, which means that adjacent frames share similarities in terms of content, motion, and appearance. Video super-resolution methods can exploit this temporal information to improve the accuracy of upscaled frames such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are two types of deep learning architectures that have proven to be effective for various computer vision tasks, including image and video super-resolution (Chadha et al.,2020). We talk about more on spatial and temporal information in a separate section later.

To further improve the quality of resolved video frames, researchers have introduced adversarial learning techniques. Adversarial networks, like the Generative Adversarial Network (GAN), consist of a generator network that produces high-resolution frames and a discriminator network that attempts to distinguish between these generated frames and ground truth high-resolution frames. Video Super-Resolution Generative Adversarial Network (VSRGAN) is an example of such a model that utilizes GAN-based architecture to generate captivating high-resolution frames (Gopan et al., 2018) [18]. To tackle the challenge posed by motion and scene, changes in video sequence models based on optical flow estimation have been developed. These models estimate motion between frames by utilizing them to generate high-resolution frames. For instance, PWCNet, which stands for Pyramid, Warping, and Cost Volume Network combines a framework that uses levels of a pyramid, with a formulation based on cost volume to calculate flow in a dependable and precise manner (Gopan et al., 2018). This calculated optical flow is subsequently employed to warp and align the low-resolution frames resulting in the creation of high-resolution video frames. Thus, SR have entered a new era of improvement as now motion models are being studied to improve the quality of videos easily.

2.3 Techniques/methods used in SR tasks

Following SR techniques and methods need to be considered while doing research in computer vision:

2.3.1. Spatial & Temporal information in Super Resolution problems:

Spatial information refers to the physical location or arrangement of objects in space, while temporal information refers to changes that occur over time. In the context of image processing, spatial information is concerned with the properties of a single

image, such as pixel values, intensity, gradient, and resolution. On the other hand, temporal information is related to a sequence of images taken at different times, such as a video. It includes changes in the object's position, size, shape, and orientation over time.

Here is an image that illustrates the difference between spatial and temporal information:

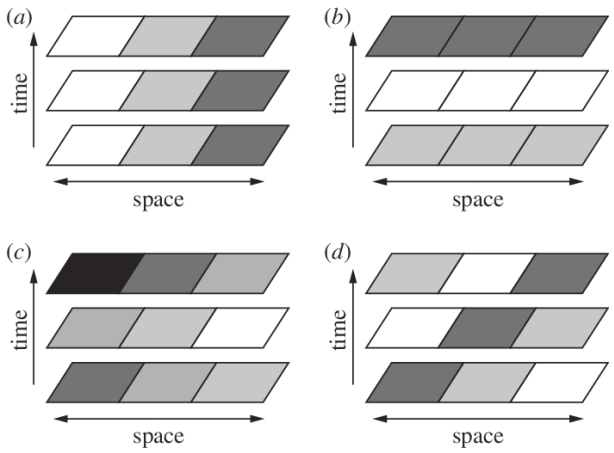

Figure 2, A graphical representation of spatial & temporal information source: White et al. *(2010)[25]*

As you can see in the image above, spatial information is concerned with the properties of a single image, such as its size and resolution. Temporal information, on the other hand, is related to a sequence of images taken at different times. In this case, we can see how the object's position changes over time.

2.3.2. Techniques to extracting spatial and temporal information

in the context of super-resolution in computer vision, extracting spatial and temporal information from images and videos is crucial for enhancing the resolution of low-quality input data. Here is a list of techniques and methods commonly used for this purpose:

2.3.2.1 Spatial Information Extraction:

- **Patch-Based Methods:** These techniques divide the image into smaller patches and use the information from nneighbouringpatches to infer missing details in the target patch. Examples include non-local means and patch-based dictionaries.
- **Convolutional Neural Networks (CNNs):** Deep learning models like CNNs have shown remarkable performance in super-resolution tasks by learning hierarchical features from low-resolution to high-resolution images.
- **Sparse Coding:** This approach represents image patches using a sparse linear combination of atoms from a learned dictionary. It can capture image structures and enhance details.
- **Self-Similarity Exploitation:** Utilizing the self-similarities present in an image to reconstruct missing details. Self-exemplars and self-similarity matrices are commonly used.
- **Edge Prior Incorporation:** Incorporating edge information as a prior to guide the super-resolution process. Edge-preserving regularization methods like total variation and gradient priors fall under this category.

2.3.2.2 Temporal Information Extraction (Video):

- **Optical Flow Estimation:** Optical flow methods compute the displacement of pixels between consecutive frames, helping to estimate motion and object trajectories.
- **Frame Interpolation:** Temporal information can be extracted by generating intermediate frames between existing frames. Deep learning techniques like

Long Short-Term Memory (LSTM) networks or Temporal Generative Adversarial Networks (TGANs) can be used for this purpose.

- **Motion Compensation:** Utilizing motion vectors to align frames and mitigate motion blur, which is crucial for improving the quality of interpolated frames.
- **Temporal Super-Resolution Networks:** These networks exploit the temporal dependencies between frames in a video sequence to enhance the temporal resolution.
- **Video Deblurring Techniques:** Addressing motion blur and camera shake in video frames using deblurring algorithms, which indirectly enhance temporal information.

It's important to note that the field of super-resolution in computer vision is rapidly evolving, and new techniques are constantly being developed.

2.4 Datasets for Super Resolution

According to a study conducted by Wei et al. (2022), researchers commonly utilize datasets to train and assess learning models regarding SR. These datasets play a role in training and validating models by providing a collection of low-resolution (LR) and high-resolution (HR) image pairs for learning. One of the employed datasets in the field of Single Image Super Resolution (SISR) is the Set5 dataset (He et al., 2019). It contains five images from categories, including nature, buildings, and textures. Another used dataset is Set14, which consists of 14 images with characteristics. These datasets are relatively small in size making them ideal for experimentation and benchmarking purposes. For evaluation, researchers often turn to larger datasets like BSD100 (Berkeley Segmentation Dataset) and the DIV2K dataset (Ju et al., 2023).

The BSD100 dataset comprises 100 images while the DIV2K dataset offers 800 training images along with 100 validation images. These larger datasets provide a sample of real-world images enabling better generalization capabilities, for trained models. According to research by Y. Wang et al. (2019), these standard datasets are used for the construction of specific datasets, which are used for the domains or tasks. For instance, in the field of medical imaging, most researchers usually create datasets through the use of high-resolution medical images and then download and sample them to mimic the LR images (Wang et al., 2022). The finding of the study, therefore, indicates that it should be permitted, and therefore, this technique permits them to be used to develop the super-resolution of the model, which is tailored to certain requirements of medical imaging applications. As noted by the study conducted by Wang et al. (2019), this makes them unique and the best materials for the creation of images and videos needed for specific use.

Most commonly used datasets in ISR and VSR tasks:

Name	*Description*	*Reference*
Set5	A dataset of 5 images commonly used for benchmarking SR algorithms	Bevilacqua et al., 2012 [52]
Set14	A dataset of 14 images commonly used for benchmarking SR algorithms	Zeyde et al., 2010 [53]
BSD100	A dataset of 100 images commonly used for benchmarking SR algorithms	Martin et al., 2001 [54]
Urban100	A dataset of 100 urban scene images with complex textures and structures	Huang et al., 2015 [55]
DIV2K	A dataset of 900 high-resolution images commonly used for training deep learning-based SR models	Agustsson and Timofte, 2017 [56]
Manga109	A dataset of 109 Japanese manga comics commonly used for benchmarking SR algorithms for manga images	Matsui et al., 2017 [57]
CelebA	A dataset of 202,599 celebrity faces commonly used for training deep learning-based SR models for face images	Liu et al., 2015 [58]

Flickr2K	A dataset of 2650 high-resolution images commonly used for training deep learning-based SR models	Lim et al., 2017 [8]
NTIRE	A series of challenges that provide datasets and evaluation metrics for various tasks in computer vision, including SR	NTIRE Workshop [60]
PIRM	A series of challenges that provide datasets and evaluation metrics for various tasks in computer vision, including SR	PIRM Challenge [61]
Vid4-4x	A dataset of 4 video sequences with low frame rate and low resolution, commonly used for benchmarking video super-resolution algorithms	Nah et al., 2017 [63]
REDS	A dataset of video sequences with low frame rate and low resolution, commonly used for training deep learning-based video super-resolution models	Nah et al., 2019

Table 1:List of Datasets found in many research papers

2.5 Comparison of Super Resolution Methods

According to a study conducted by Armannsson et al. (2021) [1], it is found that comparing methods are required to enhance image resolution where it is crucial to take into account their strengths and weaknesses. Bicubic interpolation although simple and efficient often leads to blurry and distorted results. Moreover, patch-based techniques excel in restoring texture and fine detail that helps to introduce blocking effects (Yan et al., 2023) [47]. Sparse coding and dictionary learning methods can achieve satisfying outcomes that come with a computational cost. Deep learning models and convolutional Neural Networks (CNNs) have demonstrated effectiveness in generating visually captivating high-resolution images. They are capable of learning mappings between resolution (LR) and high-resolution (HR) image pairs capturing complex nonlinear relationships (Armannsson, et al., 2021). However, these models require training data and computational resources during the training process while also needing regularization techniques to prevent overfitting. Furthermore, the performance of resolution models can vary depending on the characteristics of the input LR images. Therefore, it is

essential to choose or augment the training dataset to ensure that the model can generalize across various domains and image types.

2.6 About the base models for our research

2.6.1 An overview on EDVR model:

The EDVR model has garnered substantial attention and recognition as a foundational model in the Video Super-Resolution (VSR) task due to its exceptional performance and innovative design. To comprehensively explain its significance, we can draw a comparison with other prevalent VSR models and highlight EDVR's distinctive features that contribute to its foundational status.

EDVR stands out in the VSR landscape due to its multi-frame alignment and restoration strategy. Traditional single-frame approaches often struggle to capture temporal information and fail to restore details consistently across frames. In contrast, EDVR capitalizes on the potential of temporal coherence by incorporating a temporal alignment module that effectively handles motion and deformation within video sequences. This module integrates optical flow estimation to establish accurate correspondences between frames, enabling the aggregation of information from multiple frames for more robust restoration.

In comparison to earlier models like SRCNN (Super-Resolution Convolutional Neural Network) or FSRCNN (Fast Super-Resolution Convolutional Neural Network), which mainly focused on single-image super-resolution, EDVR's utilization of spatio-temporal information sets it apart. This enables EDVR to exploit temporal redundancies present in videos, resulting in enhanced restoration quality and temporal consistency.

Furthermore, the architecture of EDVR incorporates attention mechanisms and a feature fusion network. These components allow the model to allocate varying degrees of importance to different frames and spatial regions, attending to crucial details while suppressing noise and artefacts. By intelligently fusing features from multiple frames, EDVR excels in reconstructing high-resolution frames with finer details and reduced artefacts, which is pivotal in achieving perceptually pleasing results.

In summary, the foundational status of the EDVR model in the VSR task is substantiated by its ability to effectively harness temporal information, leverage multi-frame alignment, and employ attention mechanisms to achieve superior restoration results. Its comparison with other models underscores its unique design choices and superior performance, as demonstrated by quantitative evaluations in the research literature.

Table 1. Quantitative comparison on **Vid4** for 4× video SR. Red and blue indicates the best and the second best performance, respectively. Y or RGB denotes the evaluation on Y (luminance) or RGB channels. '*' means the values are taken from their publications.

Clip Name	Bicubic (1 Frame)	RCAN [52] (1 Frame)	VESPCN* [2] (3 Frames)	SPMC [37] (3 Frames)	TOFlow [48] (7 Frames)	FRVSR* [32] (recurrent)	DUF [10] (7 Frames)	RBPN* [6] (7 Frames)	**EDVR (Ours)** (7 Frames)
Calendar (Y)	20.39/0.5720	22.33/0.7254	-	22.16/0.7465	22.47/0.7318	-	24.04/0.8110	23.99/0.807	**24.05/0.8147**
City (Y)	25.16/0.6028	26.10/0.6960	-	27.00/0.7573	26.78/0.7403	-	**28.27/0.8313**	27.73/0.803	28.00/0.8122
Foliage (Y)	23.47/0.5666	24.74/0.6647	-	25.43/0.7208	25.27/0.7092	-	**26.41/0.7709**	26.22/0.757	26.34/0.7635
Walk (Y)	26.10/0.7974	28.65/0.8719	-	28.91/0.8761	29.05/0.8790	-	30.60/0.9141	30.70/0.909	**31.02/0.9152**
Average (Y)	23.78/0.6347	25.46/0.7395	25.35/0.7557	25.88/0.7752	25.89/0.7651	26.69/0.822	27.33/**0.8318**	27.12/0.818	**27.35**/0.8264
Average (RGB)	22.37/0.6098	24.02/0.7192	-/-	24.39/0.7534	24.41/0.7428	-/-	25.79/**0.8136**	-/-	**25.83**/0.8077

Table 2. Quantitative comparison on **Vimeo-90K-T** for 4× video SR. '†' means the values are taken from [48]. '*' means the values are taken from their publications.

Test on	Bicubic (1 Frame)	RCAN [52] (1 Frame)	DeepSR† [19] (7 Frames)	BayesSR† [21] (7 Frames)	TOFlow [48] (7 Frames)	DUF [10] (7 Frames)	RBPN* [6] (7 Frames)	**EDVR (Ours)** (7 Frames)
RGB Channels	29.79/0.8483	33.61/0.9101	25.55/0.8498	24.64/0.8205	33.08/0.9054	34.33/0.9227	-/-	**35.79/0.9374**
Y Channel	31.32/0.8684	35.35/0.9251	-/-	-/-	34.83/0.9220	36.37/0.9387	37.07/0.9435	**37.61/0.9489**

Table 3. Quantitative comparison on **REDS4**. **Left**: 4× Video SR (clean); **Right**: Video deblurring (clean). Test on RGB channels.

Method	Clip_000	Clip_011	Clip_015	Clip_020	Average
Bicubic	24.55/0.6489	26.06/0.7261	28.52/0.8034	25.41/0.7386	26.14/0.7292
RCAN [52]	26.17/0.7371	29.34/0.8255	31.85/0.8881	27.74/0.8293	28.78/0.8200
TOFlow [48]	26.52/0.7540	27.80/0.7858	30.67/0.8609	26.92/0.7953	27.98/0.7990
DUF [10]	27.30/0.7937	28.38/0.8056	31.55/0.8846	27.30/0.8164	28.63/0.8251
EDVR (Ours)	**28.01/0.8250**	**32.17/0.8864**	**34.06/0.9206**	**30.09/0.8881**	**31.09/0.8800**

Method	Clip_000	Clip_011	Clip_015	Clip_020	Average
DeblurGAN [16]	26.57/0.8597	22.37/0.6637	26.48/0.8258	20.93/0.6436	24.09/0.7482
DeepDeblur [27]	29.13/0.9024	24.28/0.7648	28.58/0.8822	22.66/0.6493	26.16/0.8249
SRN-Deblur [39]	28.95/0.8734	25.48/0.7595	29.26/0.8706	24.21/0.7528	26.98/0.8141
DBN [34]	30.03/0.9015	24.28/0.7331	29.40/0.8878	22.51/0.7039	26.55/0.8066
EDVR (Ours)	**36.66/0.9743**	**34.33/0.9393**	**36.09/0.9542**	**32.12/0.9269**	**34.80/0.9487**

Table 2:Quantitative results of EDVR Source: (Wang et al., 2019 pp:6)

For its unique architecture, it is used and mentioned by many recent literatures like (Huang & Chen, 2022)[38] and (Lu *et al.*, 2023)

2.6.1.1 EDVR: Strengths and Weaknesses

Enhanced Deep Video Restoration (EDVR) is a learning model specifically created for enhancing the resolution of videos. It utilizes both the spatial and temporal information to produce high-quality video outputs with resolution. The key advantage of EDVR lies in its capability to effectively process video sequences by taking advantage of the coherence between frames. By considering neighbouring frames EDVR can accurately estimate details and textures resulting in more realistic enhanced videos. Moreover, EDVR incorporates a motion compensation module that aligns the flow further enhancing the consistency of the output videos, over time. The potential weaknesses include computational complexity where deep learning models such as EDVR can be computationally intensive, especially when processing high-resolution video sequences. Training and inference can require significant computational resources that can limit their practicality in real-time applications or on resource-constrained devices. Another weakness is the performance of deep learning models that heavily depends on the quality and diversity of the training data. If the training dataset for EDVR is not representative of all possible scenarios and types of videos, then the ability of model is to generalize the new, unseen data that can be compromised. Thirdly, artefact amplification is another weakness of video restoration models that include EDVR that run the risk of amplifying existing artefacts present in low-quality video frames. Moreover, though the EDVR leverages temporal coherence between frames, it can struggle with scenes that involve complex motion, occlusions, or rapid changes. In such cases, it is important to align frames and estimate motion accurately can be challenging and potentially leads to suboptimal results. Lastly, deep learning models that include EDVR can be prone to overfitting if not properly regularized during training.

Hence, these are the weaknesses and strengths of using these models for SR that are proactive and efficient, but they also have their limitations that are to be addressed by future researchers.

2.6.2 An overview on YOLOv8-seg model:

To facilitate our research objective we have chosen the YOLOv8-seg model which is one of the variants of the YOLOv8 model. There are many research papers that praise the YOLOv8 model. Here is a comparison between the two latest and most powerful Segmentation models (SAM vs. YOLOv8-seg):

Model	Size	Parameters	Speed (CPU)
Meta's SAM-b	358 MB	94.7 M	51096 ms/im
YOLOv8n-seg	**6.7 MB** (53.4x smaller)	**3.4 M** (27.9x less)	**59 ms/im** (866x faster)

Table 3: YOLOv8 model's promising results. source: Ultralytics), YOLOv8 Docs (*(2023)*[2]

The paper by Reis *et al.* (2023) demonstrates that the YOLOv8 model can achieve a mAP50-95 of 0.685 and an average inference speed of 50 fps on 1080p videos.

Another research paper titled "Fast Segment Anything" by (Zhao et al., 2023) [50] proposes a new method for real-time segmentation of objects using YOLOv8-seg. There the authors says, the YOLOv8-seg model is designed specifically for semantic segmentation tasks and incorporates several new features such as **channel attention**, **spatial attention**, and **context aggregation**. The paper claims that the proposed method can achieve real-time performance without compromising on performance quality.

In addition to these papers, there are several other research papers that discuss the YOLOv8 architecture and its variants. For example, a comprehensive review paper titled "A Comprehensive Review of YOLO: From YOLOv1 and Beyond" by Terven and Cordova-

Esparza (2023b)[] provides an in-depth analysis of the evolution of the YOLO framework up to YOLOv8 and its variants.

2.6.2.1 YOLOv8-seg: Strengths and Weaknesses

YOLOv8-seg is a learning model that combines the best features of YOLOv3 and DeepLabv3+ to excel in both object detection and semantic segmentation tasks (Hussain, 2023). This model has demonstrated results in achieving resolution by leveraging the contextual information provided by semantic segmentation. One of the strengths of the variant of YOLOv8, YOLOv8-seg is its ability to accurately detect and segment objects in low-resolution images, which ensures that object boundaries are preserved during the super-resolution process. Moreover, YOLOv8-seg effectively handles video sequences by incorporating information resulting in temporal consistency, in the super-resolved outputs. However, there are drawbacks, to YOLOv8-seg. Firstly the effectiveness of the model depends greatly on how it detects objects and performs segmentation. If there are any inaccuracies in these processes, it can result in errors during the super-resolution phase. Secondly, YOLOv8-seg might face challenges when dealing with scenes that involve objects or occlusions since its main focus is, on object-level information. It is also noted that the computational complexity of YOLOv8-seg can be high, therefore, this makes it very difficult to deploy in real-time applications (Zhao et al., 2023). Therefore, it makes the best video sequence and resolution, which can be used to capture various images and perform different tasks as well.

Chapter 3: Problem Statement

The focus of this survey is two-folds. First, a basic survey on SR models and the selection of base models for the second step of this study. The second part will take care of developing a

image resolution model that can effectively improve the quality and analysis of images, in computer vision applications.

Super Resolution techniques are important in enhancing the quality of images and enabling advanced analysis. However traditional methods, for handling video image resolution problems, have many limitations. Hence investigating learning models, like VSRnet [66], SRCNN, BasicVSR [67], GAN-based model, EDVR and other video and image related models, like YOLOv8, SAM, etc. allows us to gain insights into how they perform and their potential, for tackling these issues.

3.1 Challenges and Limitations of Traditional Approaches

Conventional methods for enhancing image resolution often use interpolation techniques like interpolation to enlarge low-resolution images (Guo et., 2021). However, these methods are limited in their ability to capture the details and textures found in high-resolution images. Furthermore, they struggle with handling image structures such as edges and textures leading to unrealistic results (Guo et al., 2021). Moreover, these approaches are not well suited for enhancing video sequences as they do not effectively utilize the information that's essential for accurate resolution enhancement.

The survey's primary objective is the exploration of image resolution enhancement models to improve image quality and analysis in computer vision applications. The focus of the survey is to address limitations of traditional methods through advanced techniques like YOLOv8-seg and EDVR. It is found that traditional approaches, which rely on interpolation fail to capture high-resolution details, struggle with image structures, and are inadequate for video sequences. YOLOv8-seg excels in object detection and segmentation that leverage contextual

information for effective super-resolution and maintain object boundaries in low-resolution images. It demonstrates strength to handle video sequences along with facing challenges in accuracy and object occlusions, alongside computational complexity. On the other hand, EDVR specializes to enhance video resolution by the use of temporal coherence between frames and motion compensation that offer high-quality, realistic results. Nonetheless, EDVR counters weaknesses such as computational intensity, data diversity dependence, artefact amplification, motion complexities, and overfitting risks. Hence, these models show promise to enhance super-resolution efficiency, but their strengths and limitations warrant attention for future research refinement.

Chapter 4: Research Objectives

The primary aim of the study is to conduct a survey on learning models used for achieving super-resolution (SR) in images. The study is motivated to explore the advancements and techniques in learning models that can enhance image resolution and quality. The study will examine the techniques employed in these models, how the contributors evaluate their performance and identify their limitations. Through this analysis, the aim is to gain insights into the strengths, weaknesses, and areas for improvement of models.

In addition, the research aims to propose, implement, and evaluate a novel image resolution model. The study will also explore techniques and existing models that can enhance image quality through SR. The focus will be on developing efficient models, with improved generalization and robustness capabilities. As well as, the study intends to undertake an investigation into the possibility of boosting video SR performance by using two models which are possibly YOLOv8-seg and EDVR.

This study shall, therefore, involve the extraction and integration of the different videos with the aim of discovering the way these models can complement each other and can be used for video SR. This paper will talk about this more in the following chapter.

Chapter 5: Research Methodology

5.1 Overview

The research methodology is composed of a comprehensive investigation of techniques to enhance image and video resolutions. The focus of the study is on both classical algorithms and deep learning-based methods. The primary objective is the analysis of the effectiveness of these techniques in improving image quality and resolution through the consideration of the advancements brought forth by neural networks and computational capabilities. The research journey commences as it delves into classical algorithms that include nearest neighbour interpolation and bicubic interpolation and many other aspects as it appears. These techniques serve as fundamental tools for comparison. Moreover, these techniques are widely employed in image processing and provide a basis to evaluate the advancements introduced by modern approaches.

However, the research places a prominent stress on learning-based methods, which have gained prominence in the last few years due to the revolutionary strides made in neural networks and computational power. The Convolutional Neural Network (CNN) emerges as a pivotal architecture for super-resolution (SR) that offers the capability for learning intricate mappings between low-resolution and high-resolution images. This further leads to the generation of visually compelling outputs. This study specifically probes into the application and evaluation of

CNN-based models that include the Super-Resolution Convolutional Neural Network (SRCNN) and Deep Super Resolution (VDSR). The implementation and training of these machine learning models are facilitated by powerful frameworks that are TensorFlow and PyTorch. These frameworks provide a rich array of resources and functionalities for model design and training.

In pursuit of a robust and efficient research endeavour, the study exploits the pre-trained models and techniques that have noticeably leveraged the transfer learning for expediting the training process and enhancing the overall performance of the models. In order to assess the effectiveness of the proposed resolution models, the research has employed rigorous metrics that include the Peak Signal Noise Ratio (PSNR) and Structural Similarity Index (SSIM). PSNR gauges the discrepancy between the high-resolution and generated images whereas SSIM offers a comprehensive assessment through consideration of both details and perceived image quality (Bashir et al., 2021). This multifaceted evaluation framework is a source of ensuring a thorough analysis of the resolution enhancement techniques that sheds light on their strengths and limitations.

Expansion beyond image resolutions, the research extends its focus to video super-resolution that is a more intricate task that demands the amalgamation of information for generating high-quality frames. In this pursuit, the research probes into specialized deep learning architectures tailored for video super-resolution that is exemplified by the Enhanced Deep Video Restoration (EDVR) model. EDVR connects spatial information for elevating video frame quality and consistency is a critical aspect in video enhancement where the implementation of resolution models for videos involves the processing of sequential frames to produce videos with enhanced resolution and seamless visual continuity (Zhang et al., 2023).

So, the research methodology is anchored in a systematic investigation of diverse techniques to enhance image and video resolutions in qualitative manner. This methodology is encompassed on a multi-step process that spans from foundational algorithms to advanced deep learning models. The overarching goal is to unravel the effectiveness of these techniques while contributing to the field of computer vision to address the challenges related to image and video quality improvement. This research commences to lay the groundwork with classical algorithms that include nearest neighbour interpolation and bicubic interpolation where these methods serve as a crucial technique that enables a clear comparison between traditional techniques and the cutting-edge approaches under scrutiny.

Notably, the nearest neighbour interpolation involves replication of the nearest pixel's value to fill in gaps, while bicubic interpolation employs cubic polynomials in the estimation of pixel values. These techniques, though basic technique provide a reference point to evaluate the strides made by more intricate methods in terms of generating realistic and high-quality outputs. However, the heart of the research lies in the exploration of learning-based methods that have changed the landscape of resolution enhancement that is powered by the advent of deep learning and accelerated computational resources. Convolutional Neural Networks (CNNs) take centre stage as a prime architectural choice due to their proven success in various computer vision tasks (Battleday et al., 2021). The study is about the application and evaluation of CNN-based models tailored for super-resolution especially the Super Resolution Convolutional Neural Network (SRCNN) and the Deep Super Resolution (VDSR) model. All these models are based on machine learning techniques that help to know about the improvements in imaging along with research gap that needs to be mitigated. So, to translate these machine learning models into tangible results, there is a need to use frameworks such as TensorFlow and PyTorch. These

platforms offer a rich array of tools for designing, training, and fine-tuning machine learning models that allows the researchers to effectively implement and experiment with various architectures. However, the true test of these techniques' efficacy lies in their evaluation, which is achieved through rigorous metrics that gauge both the quantitative and qualitative aspects of resolution enhancement. The Peak Signal Noise Ratio (PSNR) is a measure of the average error between the high-resolution reference image and the generated image that provides insight into the fidelity of the generated output (Mahmoud & Kang, 2023). The Structural Similarity Index (SSIM) offers a more holistic assessment that focuses in details, luminance, and structural coherence that reflects the perceptual quality of the enhanced images (Tang et al., 2023). Expanding the research horizon to video super-resolution introduces an additional layer of complexity, demanding the fusion of temporal and spatial information. The Enhanced Deep Video Restoration (EDVR) model takes centre stage here, as it uniquely capitalizes on both the coherence between frames and motion compensation. This approach entails processing multiple frames in a sequence to generate high-resolution videos that boast visual consistency and fluidity. The research delves into the intricacies of evaluating video super-resolution models.

For evaluating the performance of the models, modern metrics such as the Video Quality Metric (VQM) and Structural Similarity based Video Quality Metric (SSIM VQM) play a pivotal role in quantifying the quality of the videos along with PSNR and SSIM. VQM measures the distortions between the original and enhanced video frames, factoring in temporal variations. On the other hand, SSIM VQM extends SSIM to the video domain, considering both spatial and temporal attributes.

In summation, the research methodology unfolds as a meticulous journey through the realms of image and video resolution enhancement. From the establishment of benchmarks

through classical algorithms to the in-depth exploration of deep learning models, from framework utilization to metric-driven evaluation, the methodology encompasses a multifaceted approach. By embracing both images and videos, and by scrutinizing various facets of quality, the research endeavours to contribute nuanced insights that hold the potential to reshape the landscape of resolution enhancement methodologies within the realm of computer vision.
In essence, the research methodology embarks on a comprehensive exploration of various techniques for achieving image and video resolutions, ranging from classical algorithms to cutting-edge deep learning approaches. The research strategy involves a systematic implementation and testing process, facilitated by state-of-the-art frameworks, with a robust evaluation framework comprising multiple metrics ensuring the thorough analysis of both image and video resolution enhancement techniques. Through this multifaceted methodology, the research aims to contribute to the advancement of resolution enhancement methods and provide valuable insights into their practical implications across diverse computer vision applications.

5.2 Methodology breakdown

In this study we are following qualitative research approach.

The first part goes with the investigation on the methods, SR models and techniques followed by the authors of YOLOv8 and EDVR to fulfil our research objective. Following this, we examine the architecture of SEEM model from (Lu et al.)[40] and Improved-EDVR from Huang et al. [38] to materialize our research objective. These papers and their mentioned architecture and models are peer reviewed in the several other papers and practically implemented in real life applications.

The second part consists of an in-depth examination of the methods and techniques the authors followed in those models to achieve video/image SR.

In the last and third part, we provide a detailed explanation of the algorithms, frameworks, and tools we utilize for implementing and testing the our proposed models. The model creation, implementation and results will come in the following section of this paper.

5.2.1 Evaluation of YOLOv8 and EDVR

Although we have talked about YOLOv8 and EDVR in the previous chapter, this study expects to contribute to the existing literature and knowledge on image/video super resolution by providing more comprehensive and in-depth understanding of the strengths and weaknesses of different image/video super resolution models.

5.2.1.1 Architecture of EDVR:

The EDVR model architecture proposed in the paper (Wang et al., 2019) consists of several key components :

- **Pyramid, Cascading and Deformable (PCD) alignment module**: This module aligns frames at the feature level using deformable convolutions in a coarse-to-fine manner to handle large motions in video frames .
- **Temporal and Spatial Attention (TSA) fusion module**: This module applies attention both temporally and spatially to emphasize important features for subsequent restoration .
- **Feature extraction network**: This network extracts features from input frames using convolutional filters .

- **Enhanced Deformable Convolutional Network (EDCN)**: This network is used to restore video frames by applying additional convolutional filters to the feature maps generated by the feature extraction network.
- **Reconstruction Module**: After alignment and fusion, the features are passed through a reconstruction module to generate the final restored frame.
- **PreDeblur Module**: This is an essential component for handling motion blur in video sequences. This module is generally integrated into the architecture to improve the quality of the restored or super-resolved frames by reducing blur artefacts.

This module works in conjunction with the Pyramid, Cascading, and Deformable (PCD) alignment module and the Temporal and Spatial Attention (TSA) fusion module. It aims to remove blur by learning a more complex mapping from the blurred to the sharp domain, often leveraging multiple layers of convolutions and non-linear activations.

The inclusion of a deblurring module makes EDVR a more comprehensive solution for video restoration, as it can handle a variety of degradation issues including blur, misalignment, and low resolution.

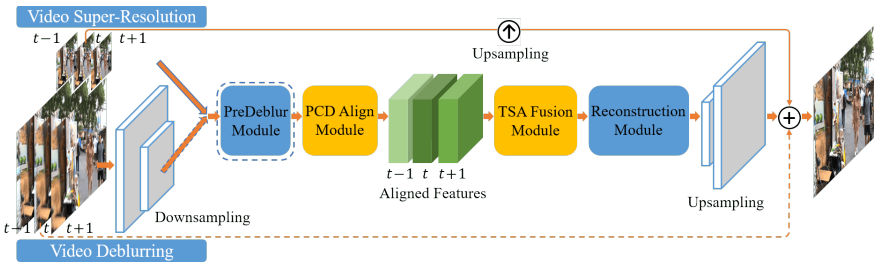

*Figure 3, source: 9*EDVR: Video restoration with enhanced deformable. *[2019]*

The other aspects of the EDVR paper are as follows:

- **Dataset**: The paper uses several benchmark datasets for training and evaluation, including REDS, Vimeo-90K, and Vid4.
- **Evaluation Metrics**: Quantitative metrics like PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index) are used to evaluate the model's performance.
- **Comparative Analysis**: EDVR is compared with existing state-of-the-art models to demonstrate its effectiveness.
- **Ablation Studies**: The authors conduct ablation studies to understand the contribution of each component in the EDVR architecture.

5.2.1.2 Architecture of YOLOv8-seg:

As we produce a hypothesis in this paper that instance segmentation could play an important role to collect features of a video or an image, we investigated YOLOv8 and its variant YOLOv8-seg in our research.

The YOLOv8 model has several enhanced capabilities compared to its predecessors, including instance segmentation, pose/keypoints estimation, and classification. Like EDVR, it is built using the PyTorch framework and provides a single Python API to work with all of its models using the same methods.

Key features of YOLOv8 model in detail:

- YOLOv8 model's variant YOLOv8-seg uses instance segmentation. This variant uses the Meta's image segmentation technique as described in their SAM model (Kirillov et al., 2023) [42]. Instance segmentation goes a step further for object detection task and involves identifying individual objects in an image and segmenting them from the rest of the image. The output of an instance segmentation model is a set of masks or contours that outline each object in the image, along with class labels and confidence scores for each object. Instance segmentation is useful when you need to know not only where objects are in an image, but also what their exact shape is.
- The model has improved accuracy and faster inference speeds than other object detection models while maintaining high accuracy. YOLOv8 supports various backbones, such as EfficientNet [68], ResNet [69], and CSPDarknet [70], giving users the flexibility to choose the best model for their specific use case.
- The architecture of YOLOv8 builds upon the previous versions of YOLO algorithms. A modified version of the CSPDarknet53 [71] architecture forms the backbone of YOLOv8.
- This architecture consists of 53 convolutional layers and employs cross-stage partial connections to improve information flow between the different layers.
- The head of YOLOv8 consists of multiple convolutional layers followed by a series of fully connected layers. These layers are responsible for predicting bounding boxes, objectness scores, and class probabilities for the objects detected in an image.
- YOLOv8 uses adaptive training to optimize the learning rate and balance the loss function during training, leading to better model performance.
- The model also employs advanced data augmentation techniques such as MixUp and CutMix [4][20][72] to improve the robustness and generalization of the model.

- YOLOv8's architecture is highly customizable, allowing users to easily modify the model's structure and parameters to suit their needs. Additionally, it provides pre-trained models for easy use and transfer learning on various datasets.

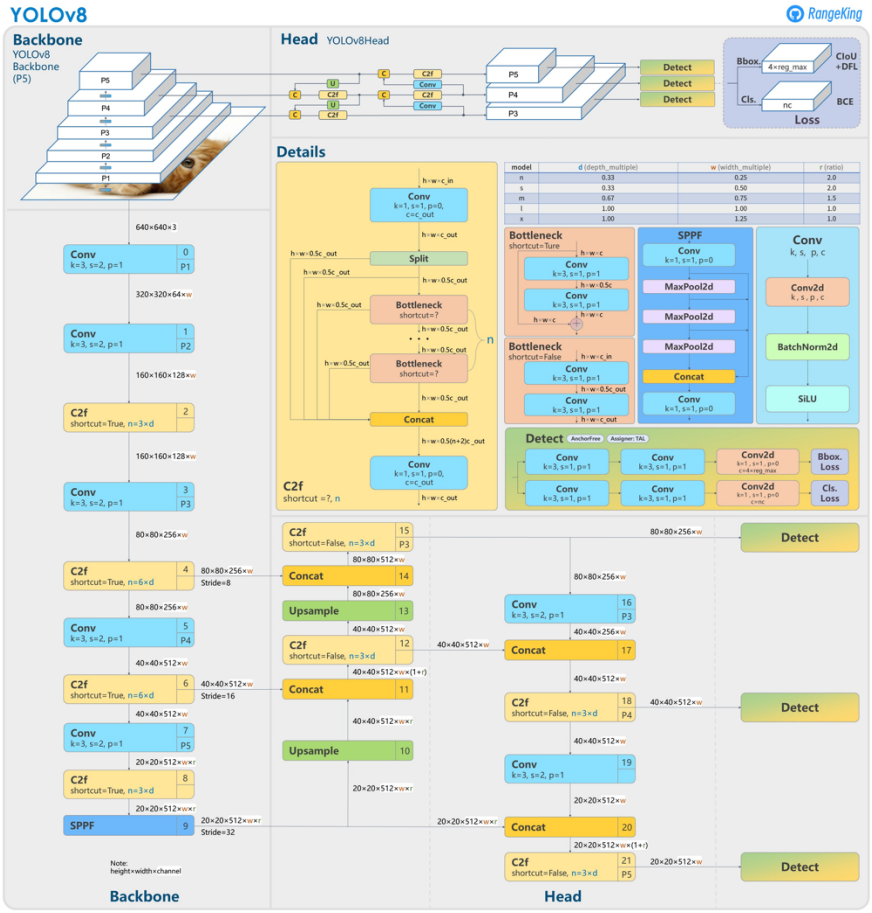

Figure 4, YOLOv8 architecture in detail.

Source: (Solawetz, What is Yolov8? the ultimate guide. *2023)[4]*

5.2.1.2.1 The detection blocks: There are 3 detection blocks in the model. The primary reasons for having multiple detection blocks in YOLOv8 are:

- **Multi-Scale Detection:** Different detection blocks operate on feature maps from different layers of the network, which inherently have different spatial resolutions. Objects of different sizes are better represented in different feature map scales. By having multiple detection blocks, YOLOv8 can simultaneously detect small, medium, and large objects in the image.
- **Hierarchical Features:** Each detection block receives features from multiple layers, including lower-level details and higher-level contextual information. This allows the network to capture both fine-grained object details and broader contextual information, leading to more accurate object localization and recognition.
- **Enhanced Performance:** By combining the outputs of multiple detection blocks, the model benefits from a richer set of features and a wider range of receptive fields. This can lead to improved detection accuracy and better generalization to different object sizes and scenes.
- **Robustness:** Multiple detection blocks provide redundancy and improve the model's robustness against various challenges, such as occlusions, object truncations, and complex scenes.
- **Feature Fusion:** Each detection block processes feature maps from different scales and layers. These feature maps are fused together through concatenation or other operations, enabling the model to leverage multi-scale features for accurate detection.

5.2.1.2.2 The 'concat' blocks: It utilizes a concatenation-based skip connection strategy to merge feature maps from different layers. This allows the model to capture both detailed and contextual information for accurate object detection and instance segmentation.

In YOLOv8-seg, the concat block manages to merge the feature maps of different layers by concatenating them along the channel dimension. Here's how the process works:

- **Feature Extraction Backbone:** The YOLOv8-seg model typically starts with a backbone network that performs feature extraction from the input image. This backbone network is usually a convolutional neural network (CNN) architecture like ResNet, Darknet, or CSPDarknet.
- **Feature Pyramid:** After the feature extraction, the model often incorporates a feature pyramid to capture multi-scale information. The feature pyramid includes multiple feature maps at different scales, where each map corresponds to a different level of the network (e.g., different layers in the backbone). These feature maps represent hierarchical features, with lower-level maps capturing detailed information and higher-level maps capturing more abstract and contextual information.
- **Concatenation of Feature Maps:** The concat block comes into play when the model needs to merge feature maps from different scales or layers. Specifically, the feature maps from lower-level layers are concatenated with the feature maps from higher-level layers. This concatenation operation is performed along the channel dimension, combining the feature maps' information.
- **Upsampling or Convolutional Layers:** After concatenation, the merged feature maps often undergo further processing. Depending on the model architecture, these merged feature maps may be upsampled (using techniques like bilinear interpolation or

transposed convolutions) to match the resolution of the target feature map. Alternatively, additional convolutional layers might be applied to refine the merged features.

- **Object Detection and Semantic & Instance Segmentation Head:** The processed and merged feature maps are then used as inputs to the object detection and semantic segmentation head of the YOLOv8-seg model. These heads generate predictions for bounding boxes, object classes, and segmentation masks.

5.2.2 An analysis and uses of feature maps:

In super-resolution research, generating feature maps is one of the important steps. Feature maps are used to extract high-level information from low-resolution images and generate high-resolution images with more detail and clarity. This is achieved by using convolutional neural networks (CNNs) to learn the mapping between low-resolution and high-resolution images. The CNNs use feature maps to extract relevant information from the input image and generate a high-resolution output image.

That said, the uses of feature maps in EDVR and YOLOv8 are different.

In the EDVR architecture, feature maps are extracted and manipulated through a series of specialized modules. It is indeed a complex process. Here's a detailed explanation:

5.2.2.1 Initial Feature Extraction:

- **Convolutional Layers**: The first step usually involves passing the input frames through one or more convolutional layers to extract initial feature maps. These layers capture basic patterns like edges, corners, and textures.
- **Activation Functions**: Non-linear activation functions like ReLU (Rectified Linear Unit) are applied to introduce non-linearity into the feature maps.
- **Pyramid, Cascading and Deformable (PCD) Alignment Module:**
- **Pyramid Levels**: The initial feature maps are processed at multiple scales in a pyramid structure. This allows the model to capture both local and global patterns.
- **Deformable Convolution**: This is a key part of the alignment module. Unlike standard convolutions, deformable convolutions allow the receptive field to adapt spatially, which is crucial for aligning features from different frames effectively. We have discussed this more in the previous chapter.
- **Cascading**: Multiple deformable alignment layers are cascaded to refine the alignment progressively.

5.2.2.2 Temporal and Spatial Attention (TSA) Fusion Module:

- **Temporal Fusion**: Feature maps from aligned neighbouring frames are fused. This is done by calculating attention weights for each time step, allowing the model to focus on more relevant frames.
- **Spatial Attention**: Within each feature map, spatial attention mechanisms are used to weigh the importance of different regions. This is particularly useful for focusing on

areas that are more likely to be degraded.

5.2.2.3 Reconstruction of Module:

- **Convolutional Layers**: The fused and aligned feature maps are passed through additional convolutional layers for reconstruction.
- **Residual Blocks**: These may be used to capture more complex patterns and improve the quality of the reconstructed frame.

5.2.2.4 Deblurring Module:

- **Convolutional Layers**: Specialized convolutional layers are used to reduce blur. These layers are trained to map blurred regions to their sharper counterparts.
- **Non-linear Activations**: Functions like ReLU or Leaky ReLU are applied to introduce non-linearity, which helps in capturing complex deblurring patterns.

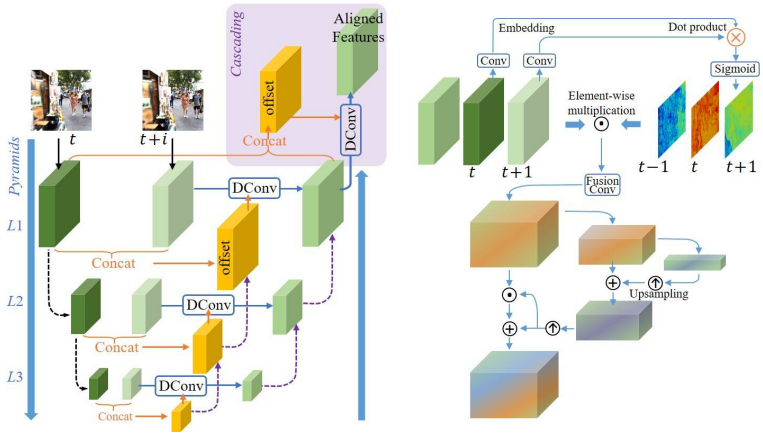

Figure 5, source: PCD(left) and TSA (right) of EDVR[64]

5.2.3 Output:

The final feature maps are transformed back into the pixel domain to produce the restored or super-resolved frame.

5.2.4 Summary:

In summary, feature extraction in YOLOv8/YOLOv8-seg involves transforming the input image into a set of abstract features that capture important visual information while discarding irrelevant details. These features are then used for making predictions in downstream tasks such as object detection and semantic segmentation.

In YOLOv8-seg, the feature extraction process mainly revolves around the use of the CSPDarknet53 backbone network. CSPDarknet53 is a modified version of the Darknet architecture that incorporates Cross-Stage Partial (CSP) connections. These connections facilitate the flow of information across different stages of the network, enabling the model to capture both low-level and high-level features effectively.

The CSPDarknet53 backbone consists of several convolutional layers organized in a hierarchical manner. It begins with a series of convolutional and pooling layers that progressively reduce the spatial dimensions of the input image while increasing the number of channels. This process helps the network capture low-level features like edges, textures, and simple shapes.

As the feature maps move deeper into the network, more complex and abstract features are extracted. The CSP connections allow the network to combine features from different stages, enabling a richer representation of the input image. This is important

for tasks like object detection and semantic segmentation, where capturing both local and global context is essential.

The feature maps obtained from the CSPDarknet53 backbone are then used as input for the object detection and semantic segmentation branches of YOLOv8-seg. These branches consist of additional layers that process the features and make predictions specific to each task.

Overall, feature extraction in YOLOv8-seg is designed to strike a balance between capturing fine-grained details and global context. The CSPDarknet53 backbone's architecture and connections play a significant role in enabling the model to learn and represent intricate features efficiently.

5.3 The foundation of EDVR is DCN

To address the limitations of Convolutional Neural Networks (CNNs) in modelling geometric transformations, the Deformable Convolutional Network (DCN) was introduced by Dai et al., 2017[3].

Their paper argues that traditional CNNs are inherently limited in handling large, unknown geometric transformations due to their fixed geometric structures. For example, a standard convolution unit samples the input feature map at fixed locations, and a pooling layer reduces the spatial resolution at a fixed ratio. These limitations are particularly problematic for tasks that require fine localization, such as object detection and semantic segmentation.

The proposed deformable modules can be easily integrated into existing CNN architectures and trained end-to-end using standard back-propagation. The authors claim that their approach is

effective for complex vision tasks and is the first to show that learning dense spatial transformations in deep CNNs can be beneficial.

They proposed two new modules: deformable convolution and deformable RoI (Region-of-Interest) pooling. These modules aim to enhance the transformation modelling capabilities of CNNs by augmenting the spatial sampling locations with additional offsets, which are learned from the target tasks without requiring additional supervision.

5.4 Factors that influence the performance and quality of image/video super-resolution models like EDVR

5.4.1 Data and Preprocessing

- **Training Data**: The quality and diversity of the training data significantly affect the model's performance.
- **Data Augmentation**: Techniques like rotation, flipping, and cropping can improve the model's generalization ability.
- **Normalization**: Proper normalization of input data can stabilize the training process.

5.4.2 Loss Functions

- **Reconstruction Loss**: Typically, L1 or L2 loss is used to measure the difference between the super-resolved and ground-truth images.

- **Perceptual Loss**: This loss measures high-level features, often using a pre-trained network like VGG, to ensure that the super-resolved images are perceptually similar to the ground truth.
- **Adversarial Loss**: In some cases, a GAN framework is employed to make the super-resolved images more realistic.

5.4.3 Training Strategies

- **Learning Rate**: The choice of learning rate and its scheduling can affect convergence speed and final performance.
- **Batch Size**: A suitable batch size can balance the trade-off between memory usage and model performance.
- **Optimization Algorithm**: Algorithms like Adam, SGD, etc., can influence the training dynamics.

5.4.4 Computational Resources

- **Hardware**: The availability of high-performance GPUs can enable the training of more complex models.
- **Memory**: Limited GPU memory can restrict the model size and training batch size, affecting performance.

5.4.5 Evaluation Metrics

- **PSNR (Peak Signal-to-Noise Ratio)**: Commonly used for quantitative evaluation but may not always correlate with human perception.

- **SSIM (Structural Similarity Index)**: Provides a better approximation of perceptual quality.
- **User Studies**: Sometimes, subjective evaluation is conducted to assess the perceptual quality of super-resolved images.

5.4.6 External Factors

- **Motion Blur**: The presence of motion blur in the video can affect the model's performance.
- **Noise Level**: Different levels of noise in the input can also be a factor.

5.5 Implementation background

As this study requires to provide practical and useful suggestions for improving the performance and quality of image/video super-resolution models, we come up with a hypothesis to improve or complement EDVR model. To facilitate our work we have inspired by and chosen two models for achieving a better image/video SR for our study. These models are SAM-guided Ed refinement Module (SEEM) by (Lu et al., 2023) and Improved EDVR by (Huang & Chen, 2022). We follow their method of work and tools to work with EDVR and YOLOv80seg pretrain models.

5.5.1 SEEM:

SEEM's plug-in module boosts the EDVR's performance and quality by using the semantic information from SAM to improve the alignment and fusion of multiple frames.

According to the paper, SEEM can enhance both the foreground and background regions of the video frames, resulting in more accurate and realistic super-resolution outputs. For example, in section 4.2 of the paper, the authors show some visual comparisons between EDVR and EDVR+SEEM on the REDS dataset. They demonstrate that SEEM can better handle large motions, occlusions, and complex textures, such as the moving car, the occluded person, and the brick wall.

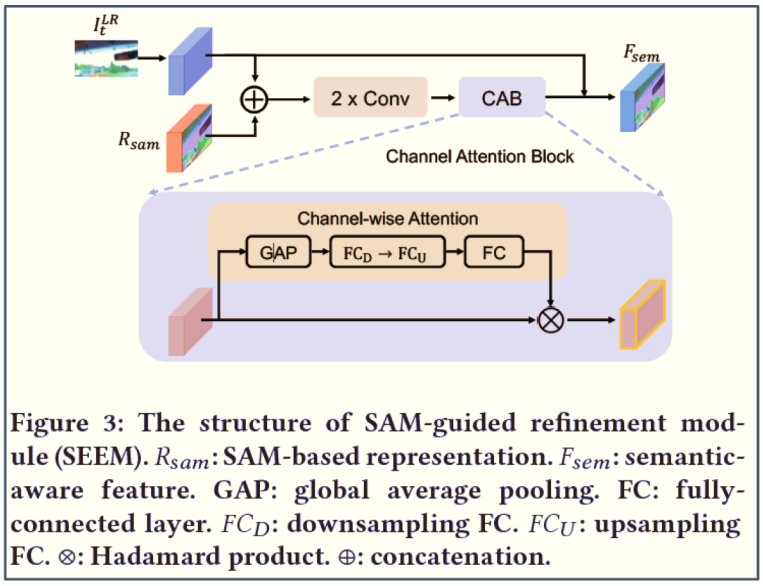

Figure 3: The structure of SAM-guided refinement module (SEEM). R_{sam}: SAM-based representation. F_{sem}: semantic-aware feature. GAP: global average pooling. FC: fully-connected layer. FC_D: downsampling FC. FC_U: upsampling FC. ⊗: Hadamard product. ⊕: concatenation.

Figure 6: SEEM model's architecture source: (Lu et al.*, 2023)*

They also provide some quantitative results in Table 2 [40] of the paper, where they report that SEEM can improve the PSNR and SSIM metrics of EDVR by 0.14dB and 0.003, respectively, on the REDS dataset.

Moreover, SEEM can also reduce the number of parameters and FLOPs of EDVR by 11.6% and 10.8%, respectively, without sacrificing performance, as shown in Table 3

[40] of the paper. Therefore, SEEM's plug-in module can boost EDVR's performance and quality by utilizing a more robust and semantic-aware prior for VSR.

5.5.2 Improved EDVR:

The authors identify two main challenges in VSR: accuracy and the need for high-speed, possibly real-time, models.

To address these challenges, the authors propose several improvements:

- Preprocessing Module: A new preprocessing module is introduced to handle blurring situations and emphasize effective features. This module consists of rigid convolution sub-modules and feature enhancement sub-modules.
- Temporal 3D Convolutional Fusion Module: This module aims to extract information from image frames more accurately and quickly.
- New Reconstruction Block: A new block is designed to better utilize the information in feature maps by introducing a new channel attention approach.
- Additionally, the authors employ multiple programmatic methods to accelerate both the model training and inference processes, making the model more practical for real-world applications.

The paper argues that traditional algorithms like bicubic and bilinear interpolation are not sufficient for achieving ideal VSR output. Machine learning-based algorithms offer better results but often come at the cost of time-consuming training and large model parameters. The proposed improvements aim to balance accuracy and efficiency, making the model more suitable for practical applications.

The proposed model introduces several innovations to improve upon the original EDVR model in terms of robustness, efficiency, and performance.

Performance Comparison with Original EDVR Model

- PSNR (Peak Signal-to-Noise Ratio): The proposed model achieved a PSNR of 29.7801, which is higher than the baseline EDVR (M) model with a PSNR of 28.7416.
- SSIM (Structural Similarity Index): The proposed model scored an SSIM of 0.8552, showing nearly a 4% improvement over the baseline EDVR (M) model, which had an SSIM of 0.8245.
- Parameter Efficiency
- The proposed model has 3,468,295 parameters, which is similar to the EDVR (M) model with 3,300,131 parameters.
- Compared to EDVR (L) with 20,633,827 parameters, the proposed model has only 16.8% of that, making it more parameter-efficient.

We assume, the proposed model not only outperforms the baseline EDVR (M) model in terms of PSNR and SSIM but also maintains a similar level of parameter count. This makes the model both robust and efficient, reducing timing cost and memory consumption while delivering extraordinary performance.

5.6 Programming tools and frameworks

Programming tools: PyTorch (*PyTorch.* Available at: https://pytorch.org) is a widely used open-source machine learning framework. It is particularly popular in the fields of computer vision and natural language processing (NLP), which aligns our research focus.

In the context of computer vision, PyTorch provides a versatile and flexible platform for developing and deploying deep learning models. It offers several key features that make it a preferred choice for computer vision tasks:

- Dynamic Computational Graphs: PyTorch uses dynamic computational graphs, which means that the graph is built and executed on the fly as operations are performed. This allows for more intuitive debugging and dynamic control over the model's behaviour, making it well-suited for research and experimentation.
- Torch-Vision: Torch-Vision is a popular library within PyTorch specifically designed for computer vision tasks. It provides a wide range of pre-processing transforms, datasets, and model architectures commonly used in image classification, object detection, segmentation, and more.
- Customization: PyTorch's dynamic nature enables researchers to easily define custom architectures, loss functions, and training loops. This level of customization is beneficial for testing novel ideas and algorithms in the field of computer vision.
- Visualization and Debugging: PyTorch integrates well with libraries like Tensor Board and matplotlib, making it easier to visualize training progress, model architecture, and intermediate outputs during the development process.

- Transfer Learning: Transfer learning is a common technique in computer vision where pre-trained models are fine-tuned for specific tasks. PyTorch provides access to various pre-trained models through its Torch-Vision library, allowing researchers to leverage the power of large-scale models trained on massive datasets.
- Research-Focused: PyTorch's design philosophy and dynamic nature are well-aligned with the iterative and exploratory nature of research. Researchers can experiment with new ideas quickly and adjust model components easily.
- Community and Ecosystem: PyTorch has a vibrant and active community that contributes to the development of libraries, tools, and resources. This ecosystem includes research papers, code implementations, and online forums where researchers share insights and expertise.

It's important to note that while PyTorch is favored by researchers for its flexibility and dynamic nature, TensorFlow, another popular framework, also offers strong capabilities in computer vision with its TensorFlow-Keras API.

Platform: There are several cloud platforms which facilitate GPU and enough RAM for this type of research. But they are often very costly to avail if a research belongs to a single person. That being said, Google Colab (https://colab.research.google.com) and Amazons AWS (https://studiolab.sagemaker.aws.com) platforms give some sort of free services. We use those whenever needed.

5.7 Conclusion

Video Super-Resolution (VSR) is a burgeoning field in computer vision that aims to enhance the resolution of video sequences. The Enhanced Deformable Video Restoration (EDVR) model serves as a foundational architecture, focusing on deformable convolutions and attention mechanisms to improve VSR performance. An improved version of EDVR introduces innovations like a Temporal 3D Fusion module and a new Reconstruction Block, achieving better performance metrics while maintaining parameter efficiency. Another notable model that leverages the Segment Anything Model (SAM) introduces a SAM-guided Ed refinement Module (SEEM) to enhance both alignment and fusion procedures in VSR. These models collectively represent the evolving landscape of VSR, each contributing unique methodologies to tackle the challenges of accuracy and computational efficiency.

Again, as this study is qualitative in nature and thus has not focus on quantitative benchmarks for model performance. Only some peer-reviewed papers are used to demonstrate their validity in the computer vision area of study. Also, the rapidly evolving nature of machine learning models may render some findings obsolete in a short period. But as of the paper published in, we can say that our outline methodology is current and very relevant to this area of research interest.

Chapter 6: The Proposed Model

We already come across that deformable convolutional networks (DCNs) improves object detection and semantic segmentation task. We also found that SAM's segmentation task gave a

tremendous boost in various tasks in the computer vision field. The super popular and practical model YOLOv8 showed us how SAM's image segmentation capability helps to achieve better results. Also, the plug-in module of SEEM showed a use-case of SAM's extracted feature maps in various ways which boost the VSR.

All of these models and techniques guide us to come up with an idea to improve the super-resolution of an image and video frames.

6.1 Our module hypothesis:

To formulate a model, we intend to use the YOLOv8-seg model's feature maps and fuse those features with the EDVR's aligned feature. We use the following approach to achieve our goal:

- First, we aim to collect all the feature maps from the image/video frame that produced by the YOLOv8-seg model.
- This will give us an idea of where the objects are located in each frame through feature masks. This give us the internal information of the feature maps like shape and size.
- Next, we can use EDVR to super-resolve each frame of the video. This will give us high-quality image frames with improved perceptual quality.
- Then, we can fuse the features extracted from YOLOv8-seg with the aligned features from EDVR to generate high-quality image or video frames with very good image segmentation.

We hypothesise that the above approach allows us to generate high-quality image/video frames with object detection masks. However, it is important to note that this approach may require significant computational resources due to the complexity of both models.

Tools: We use Python-based Pytorch as our programming language because it has better packages for the research objective. It is easy to use and is built with research tasks in mind. We, initially, use Jupyter Notebook and VSCode as IDE to investigate, debug and create our module's code. After that, we transfer our model to Google-Colab's GPU for faster output.

Dataset: We came across many datasets that have been used in computer vision tasks; especially ISR and VSR tasks. We mentioned the details of those datasets in the literature review section. In our task, we aim to use a small dataset of image frames from VID4, Set5 or Set14 and possibly from REDS dataset. We aim to carry on our tasks on a single image first then go on with collection of video image frames.

About Result: As this study is qualitative, our primary aim to produce result from this proposed model are images. Then we compare those with the input images and ground truth images provided by the well-known image frames of our selected datasets.

So, qualitative visual comparisons between different resolution models are important part of this study.

Our secondary aim is to produce the mathematical measuring techniques like PSNR and SSIM if we can have enough time. There been always a time constraint in this type of study to get a full-fledged outcome.

6.1.1 A graphical view of the proposed model

This graphical view of the proposed model could be changes after the experimentation and obtained results.

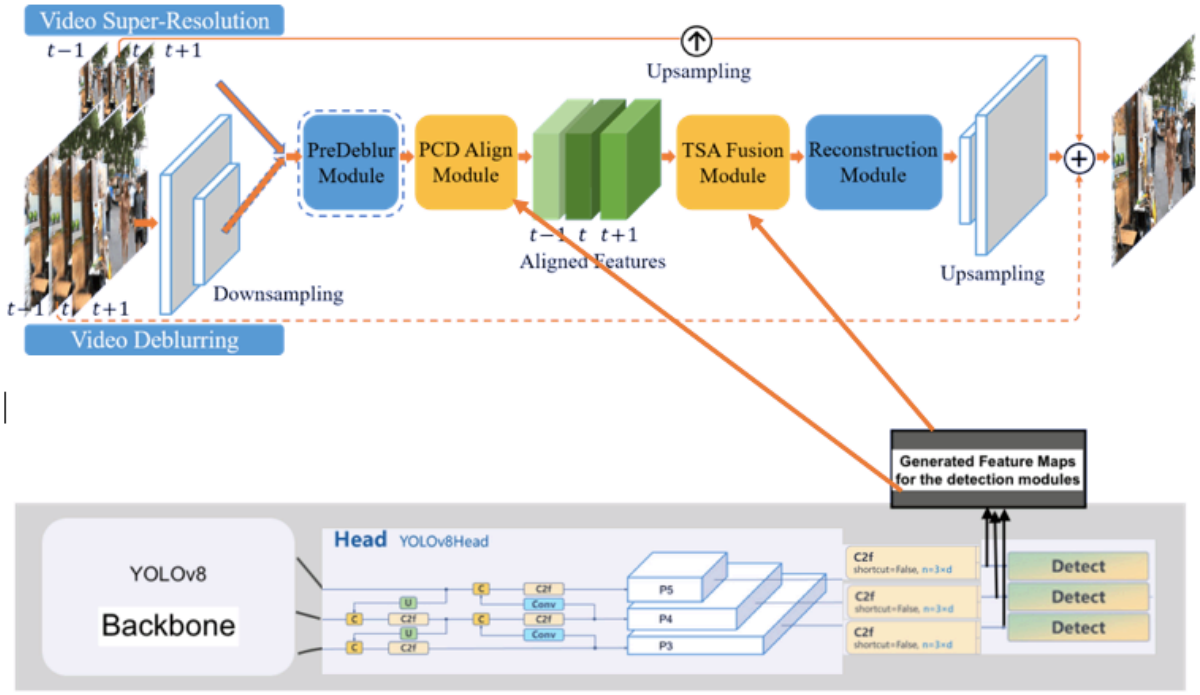

Figure 7, a graphical representation of the proposed model, top structure is the EDVR model and bottom one the YOLOv8 model

The feature maps generated from the YOLOv8 model will be concatenated into the PCD alignment module or/and the TSA fusion module. The end result will be obtained from the EDVR model.

We experiment this by writing codes using Pytorch on the local machine with CPU and with single GPU from Google's Colab environment. Our initial aim is to set up a workable test and train environment and then develop code to make our proposed model to fit in the EDVR model. In this case, the main architecture file of the EDVR model will be customized.

Chapter 7: Experiments, Result and discussion

Here, we layout our experiments, result and discussion in detail.

7.1 YOLOv8-seg uses:

Initially we start to implement the YOLOv8-seg pretrained model to obtain and examine the feature mask from the 3 outputs of the final 3 c2f modules. All the c2f module are 3x3 conv.

We use Jupyter Lab and VSCode on local machine. We found the YOLOv8 document is clear enough to carry out this task. The creators and maintainer (Ultralytics, 2023) make this easy for everybody.

We use instance segmentation from the pretrained model (yolov8s-seg) for our task. Here is a list of supported task:

Model Type	**Pre-trained Weights**	**Task**
YOLOv8	yolov8n.pt, yolov8s.pt, yolov8m.pt, yolov8l.pt, yolov8x.pt	Detection
YOLOv8-seg	yolov8n-seg.pt, yolov8s-seg.pt, yolov8m-seg.pt, yolov8l-seg.pt, yolov8x-seg.pt	Instance Segmentation
YOLOv8-pose	yolov8n-pose.pt, yolov8s-pose.pt, yolov8m-pose.pt, yolov8l-pose.pt, yolov8x-pose.pt, yolov8x-pose-p6.pt	Pose/Keypoints
YOLOv8-cls	yolov8n-cls.pt, yolov8s-cls.pt, yolov8m-cls.pt, yolov8l-cls.pt, yolov8x-cls.pt	Classification

Table 4:YOLOv8's different versions. Source: (Ultralytics, 2023)[3]

But, to better understand and tuning the hyperparameter, we also used the YOLOv8-seg.yml file like this "**model = YOLO('yolov8n-seg.yaml')**". Then we trained it on COCO128 dataset like this:

```
[34]: # Train the model
      results_yml = ymodel.train(data='coco128-seg.yaml', epochs=3)
```

Figure 8, Training with YOLOv8-seg.yml

Some results from these are as follows:

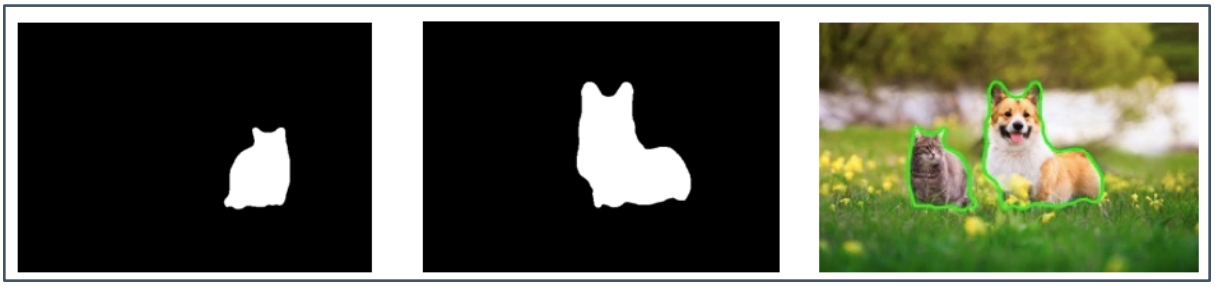

Figure 9, Feature masks from YOLOv8-seg

7.2 EDVR uses:

Because of the old and complex documentation, implementing EDVR model was tedious. For investigating the model, the simplest way to use the model are as follows:

1. Install and update basic environmental packages from Python and Pytorch's torch and torchvison.
2. Clone the latest EDVR packages from BasicSR found in Github repository (Xinntao, *Xinntao/EDVR: EDVR has been merged into BASICSR.*) [65]

3. The following should be done in order to install EDVR model as shown in this image:

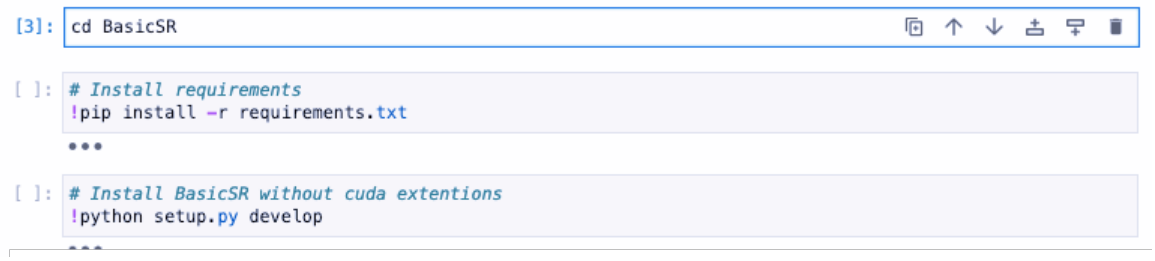

Figure 10, EVDR installation

4. Train and test can be done by keeping the datasets on disk which is the easiest way or can be done other ways mentioned in their Github repo (link given earlier)
5. We use image frames from the datasets Set4 and REDS. Training and testing required greater computational power and time on the CPU. Even with a single GPU, it could require hours of time. We use Google's Colab environment.
6. The results are impressive as stated in the original paper. Some results are here which are found the same in our experiments:

Figure 11: Results from the EDVR project. source: (Xintao, EDVR project*)[58]*

7.3Developing our proposed module:

7.3.1 Feature extraction from YOLOv8:

As we plan, according to our proposed method, we need to gather feature maps from the YOLOv8-seg model. We used pretrained model for this purpose. In this case, it is straight forward as no configuration is needed or intended for our study.

We write a block of code using Pytorch to extract feature maps which are generated from the three c2f module.

```python
global_index = 0
global_feature_maps_l0 = []
global_feature_maps_l1 = []
global_feature_maps_l2 = []

def hook_fn_before_seg_head_l0(m, x, y):
    # In the case of multiple images it should return all
    for it in y:
        global_feature_maps_l0.append(it.detach())

def hook_fn_before_seg_head_l1(m, x, y):
    # In the case of multiple images it should return all
    for it in y:
        global_feature_maps_l1.append(it.detach())

def hook_fn_before_seg_head_l2(m, x, y):
    # In the case of multiple images it should return all
    for it in y:
        global_feature_maps_l2.append(it.detach())


def hook_install(layer, hook_fn):
    # Register the hook on the 5th layer of the model's 'layer1'
    layer.register_forward_hook(hook_fn)

    # Now, whenever you make a forward pass through the model, the hook will be triggered when the in

hook_install(yolo_model.model.model[21], hook_fn_before_seg_head_l2)
hook_install(yolo_model.model.model[18], hook_fn_before_seg_head_l1)
hook_install(yolo_model.model.model[15], hook_fn_before_seg_head_l0)

# Predict with the model
#results = yolo_model('https://ultralytics.com/images/bus.jpg')  # predict on an image
results = yolo_model('                                                cat_dog.jpeg')
```

Figure 12, Code example- Feature extraction from YOLOv8-seg model with a sample image

This test was successful. So, we happily processed to the next stage.

We did the same process to extract features by using Pytorch's 'hook_install' method from some image frames from the Set5 (from the calendar folder) dataset. We saved our 3 feature maps from the 3 outputs (which are used later to the detection module by YOLOv8) as Pytorch model checkpoint files (with .pt extension) in a separate directory as shown in the figure below:

```python
# Save global_feature_maps_l0-2
global_feature_maps_l0 = torch.stack(global_feature_maps_l0)
global_feature_maps_l1 = torch.stack(global_feature_maps_l1)
global_feature_maps_l2 = torch.stack(global_feature_maps_l2)

torch.save(global_feature_maps_l0, 'hooking2/global_feature_maps_l0.pt')
torch.save(global_feature_maps_l1, 'hooking2/global_feature_maps_l1.pt')
torch.save(global_feature_maps_l2, 'hooking2/global_feature_maps_l2.pt')
```

Figure 13, Saving feature maps from YOLOv8-seg model with a sample image

7.3.2 Use of EDVR model on extracted feature maps from YOLOv8:

Now, we started to implement EDVR for our goal of this study. For the sake of ease, we call the 'EDVR_arch' file directly in our code like below:

```
from basicsr.archs.edvr_arch import EDVR

edvr = EDVR(num_in_ch=3,
    num_out_ch=3,
    num_feat=64,
    num_frame=5,
    deformable_groups=8,
    num_extract_block=5,
    num_reconstruct_block=10,
    center_frame_idx=2,
    with_predeblur=True,
    hr_in=False)
```

Figure 14, edvr_arch call

We have tested rigorously to find out an optimum EDVR model for our use.

Here comes our main experimental part to use the feature maps' information from the YOLOv8-seg model and fuse/concat them in the EDVR model.

Our experiments on tweaking the EDVR's main architecture file (edvr_arch) mostly failed except one.

We have found that when the PCD module forwards its outputs for feature alignment with neighbouring features is the best place to merge the feature maps gathered from the YOLOv8-seg model. The code block we developed is shown in the figure below:

```
nbr_feat_l[0] = self.fusion0(nbr_feat_l[0], [ft0[:, i, ...]])
nbr_feat_l[1] = self.fusion1(nbr_feat_l[1], [ft1[:, i, ...]])
nbr_feat_l[2] = self.fusion2(nbr_feat_l[2], [ft2[:, i, ...]])
```

Figure 15, Fusion with the neighbouring features of EDVR model

This concept is applied to the EDSR model as well on a single image.

We incorporate two fusion module to achieve our purpose:

- **Attention Fusion Module**: Multiplies the input tensor **x** with attention maps generated from feature tensors **fts**(feature maps).
- **Stack Fusion Module**: Concatenates the input tensor **x** with attention maps and then applies a 1x1 convolution.

As the module names indicate, these are designed to perform attention-based fusion of feature maps.

We know that element-wise multiplication is more appropriate for tasks where spatial focus is important, while concatenation is better for tasks requiring richer feature representations.

Both modules are designed to be initialized at runtime based on the shape of the input tensors, which offers flexibility but may not be optimal for all use cases.

7.3.3 The final model structure:

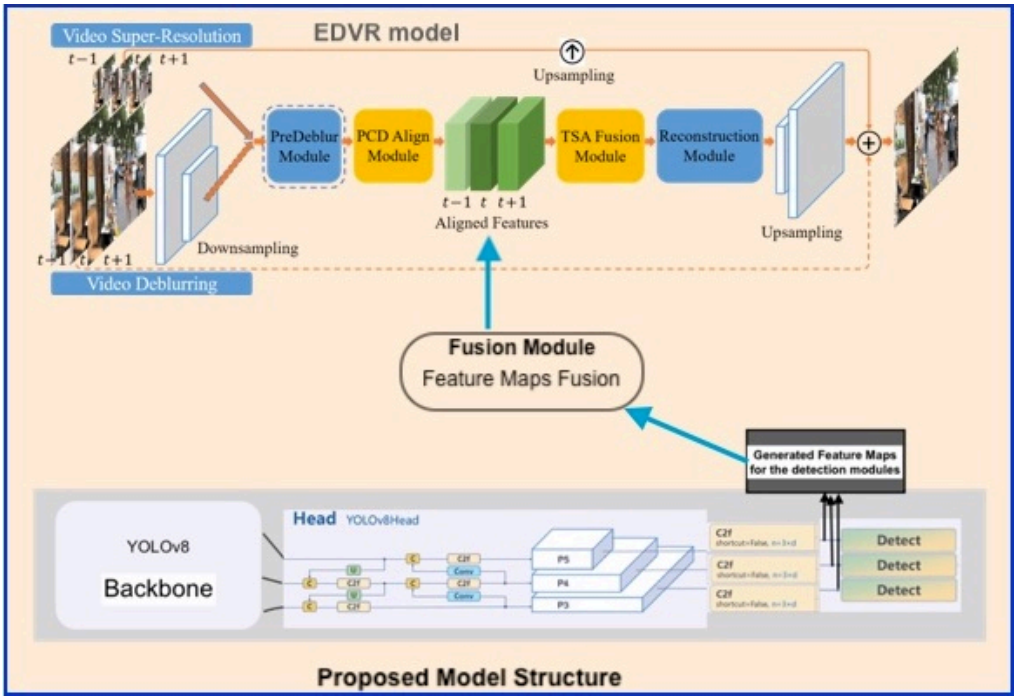

Figure 16, The final model architecture

7.3.4 Reason for the fusion models:

While experimenting the feature maps from YOLOv8-seg and EDVR models, we recognize that the models are creating and working on two different types of dimensions and sizes respectively. This resulted many considerations to develop a fusion model, like, loss of information, performance and unwanted or added information, etc.

EDVR expects input feature size containing 64 channel number of intermediate features with minimum 5 number of frames etc. While YOLOv8's generated feature map sizes are different.

For example, if the input is with a batch of 16 frames each of size 128x128 with 3 colour channels, a feature map after a convolutional layer might have a shape like **[16, 64, 128, 128]**. On the other hand, with the same batch of 16 images each of size 416x416 with 3 colour channels, we might encounter feature maps with shapes like **[16, 512, 13, 13]**, **[16, 256, 26, 26]**, and **[16, 128, 52, 52]** at different scales in the network.

7.4 Results and discussion:

1. Result in quality:

Although we had limited resources, we have found impressive results. As mentioned earlier, we conduct qualitative research and quantitative research is not our agenda in this study.

Figure 17, Comparing output with the ground truth calendar image frame from Set5 dataset

Figure 18, Comparing output with the blurred LR calendar image frame from Set5 dataset

Though our focus is not to compare, we can see this produces good quality outputs that match or even produce better quality outputs. Our model produces better quality in terms of resolution against the LR images.

The output shows some image augmentation results which can be managed via image normalization within the 'dataloader' function usage.

Also, as we have mentioned, because of the unavailability of required GPUs and time constraints, we could not carry out a full train and test tasks with the model on a full dataset.

2. PSNR result:

We have carried out a simple PSNR test on our experiment. Compared with the original paper's EDVR measures, we found a lower value. In terms of PSNR, the better value indicates better model test. So we can conclude that our proposed model gives a poorer result than the original model.

7.5 Implementation problems

The original EDVR model is very resource hungry and it takes longer times with GPUs. We must confess that our experiments were carried out in local machines with regular Intel-based CPUs and Google-Colab's GPUs which proved not good options for this entire task.

Even a PNG formatted image with a size 1280 by 720 could not be processed by the local CPU and not even Google-Colab's single GPU which provides 128MB RAM.

Overall, the architecture is a way of using two distinct computer vision models for super-resolution problems. Combining these two models on huge data of image frames will require extra computational power and resources.

Chapter 8: Conclusion

This study aimed to evaluate and propose a hybrid super-resolution model that synergizes the capabilities of YOLOv8-seg and EDVR. The primary focus was on the application of computer vision tasks, particularly in enhancing the quality of image and video frames via Super Resolution (SR) techniques.

8.1 Methodological Foundations:

The study adopted a qualitative approach and utilized Python-based PyTorch for code development. Jupyter Notebook and VSCode served as the initial IDEs, before migrating the

model to Google Colab for better computational performance. A variety of datasets such as VID4, Set5, Set14, and possibly REDS were targeted for the experiments.

8.2 Proposed Model:

Our proposition involved extracting feature maps from YOLOv8-seg and aligning these with features from EDVR to produce high-quality image frames with enhanced object detection masks. Despite the high computational requirements, the initial results showed promise. We introduced two fusion modules—Attention Fusion and Stack Fusion—to handle the different types of dimensions and sizes produced by each of the individual models.

8.3 Experimental Outcomes:

Quality of Output: The qualitative assessment demonstrated that the proposed model could produce high-quality images, superior to the Low Resolution (LR) counterparts.

Performance Metrics: A preliminary PSNR evaluation indicated that our model underperformed compared to the original EDVR model, but this could be attributed to resource and time constraints.

Challenges: Implementation was not seamless; both models have significant computational requirements, and our available resources were limited. Moreover, the complexity of EDVR's documentation presented initial hurdles.

8.4 Limitations and Future Work:

The study was constrained by limited computational resources and time. Moreover, we focused primarily on qualitative measures, leaving scope for future quantitative assessments. The model could also be tested on a wider variety of datasets and conditions.

8.5 Implications:

The fusion of YOLOv8-seg and EDVR presents a novel approach to SR, merging the

capabilities of object detection and SR uniquely. While the results are preliminary, they offer a compelling foundation for future research in computer vision.

In summary, the proposed model demonstrates the potential to advance the state of the art in super-resolution methodologies by integrating feature masks' information. However, further research is essential to fine-tune the model and evaluate its performance under different conditions and metrics.

Chapter 9: References

[1] Armannsson, S. E., Ulfarsson, M. O., Sigurdsson, J., Nguyen, H. V., & Sveinsson, J. R. (2021). A comparison of optimized Sentinel-2 super-resolution methods using Wald's protocol and Bayesian optimization. Remote Sensing, 13(11), 2192.

[2] Ultralytics (2023) *Yolov8*, *Ultralytics YOLOv8 Docs*. Available at: https://docs.ultralytics.com/models/yolov8/#usage (Accessed: 12 June 2023).

[3] Ultralytics (2023) *Architecture summary*, *Architecture Summary - Ultralytics YOLOv8 Docs*. Available at: https://docs.ultralytics.com/yolov5/tutorials/architecture_description/ (Accessed: 15 July 2023).

[4] Ultralytics (2023) *Yolov8*, *Ultralytics YOLOv8 Docs*. Available at: https://docs.ultralytics.com/models/yolov8/#usage (Accessed: 12 June 2023).

[5] Ayas, S., & Ekinci, M. (2020). Single image super resolution using dictionary learning and sparse coding with multi-scale and multi-directional Gabor feature representation. Information Sciences, 512, 1264–1278.

[6] Dai, J. *et al.* (2017) 'Deformable Convolutional Networks', *2017 IEEE International Conference on Computer Vision (ICCV)* [Preprint]. doi:10.1109/iccv.2017.89.

[7] Bashir, S. M. A., Wang, Y., Khan, M., & Niu, Y. (2021). A comprehensive review of deep learning-based single image super-resolution. PeerJ Computer Science, 7, e621.

[8] Lim, B. et al. (2017) Enhanced Deep Residual Networks for Single Image Super-Resolution, arXiv.org. Available at: https://arxiv.org/abs/1707.02921v1.

[9] Ledig, C. et al. (2016) Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, arXiv.org. Available at: https://arxiv.org/abs/1609.04802v5.

[10] Battleday, R. M., Peterson, J. C., & Griffiths, T. L. (2021). From convolutional neural networks to models of higher-level cognition (and back again). Annals of the New York Academy of Sciences, 1505(1), 55–78.

[11] Wang, X. *et al.* (2019) 'EDVR: Video restoration with enhanced deformable convolutional networks', *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* [Preprint]. doi:10.1109/cvprw.2019.00247.

[12] Bhadra, P., Balabantaray, A., & Pasayat, A. K. (2023). MFEMANet: An effective disaster image classification approach for practical risk assessment. Machine Vision and Applications, 34(5), 1–23.

[13] Chadha, A., Britto, J., & Roja, M. M. (2020). iSeeBetter: Spatio-temporal video super-resolution using recurrent generative back-projection networks. Computational Visual Media, 6, 307-317.

[14] Chen, S. (2020). Causality Inference between Time Series Data and Its Applications. Columbia University.

[15] Dong, C., Loy, C. C., & Tang, X. (2016). Accelerating the super-resolution convolutional neural network. Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14, 391–407.

[16] Geng, Z., Liang, L., Ding, T., & Zharkov, I. (2022). Rstt: Real-time spatial temporal transformer for space-time video super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 17441-17451).

[17] Guo, X., Yang, H., & Huang, D. (2021). Image inpainting via conditional texture and structure dual generation. Proceedings of the IEEE/CVF International Conference on Computer Vision, 14134–14143.

[18] Gopan, K., & Kumar, G. S. (2018, May). Video super resolution with generative adversarial network. In 2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI) (pp. 1489-1493). IEEE.

[19] He, Z., Cao, Y., Du, L., Xu, B., Yang, J., Cao, Y., Tang, S., & Zhuang, Y. (2019). MRFN: Multi-receptive-field network for fast and accurate single image super-resolution. IEEE Transactions on Multimedia, 22(4), 1042–1054.

[20] Hussain, M. (2023). YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection. Machines, 11(7), 677.

[21] Ju, R.-Y., Chen, C.-C., Chiang, J.-S., Lin, Y.-S., & Chen, W.-H. (2023). Resolution enhancement processing on low quality images using swin transformer based on interval dense connection strategy. Multimedia Tools and Applications, 1–17.

[22] Kaur, H., Koundal, D., & Kadyan, V. (2021). Image fusion techniques: A survey. Archives of Computational Methods in Engineering, 28, 4425–4447.

[23] Kim, S., Jun, D., Kim, B.-G., Lee, H., & Rhee, E. (2021). Single image super-resolution method using cnn-based lightweight neural networks. Applied Sciences, 11(3), 1092.

[24] Kim, J., Jung K. Lee, and Kyoung Mu Lee (2016). Deeply-Recursive Convolutional Network for Image Super-Resolution.

[25] White, E.P. *et al.* (2010) “Integrating spatial and temporal approaches to understanding species richness,” *Philosophical Transactions of the Royal Society B: Biological Sciences*, 365(1558), pp. 3633–3643. doi:10.1098/rstb.2010.0280.

[26] Kang, J. *et al.* (2020) “Deep Space-Time video upsampling networks,” in *Springer eBooks*, pp. 701–717. Available at: https://doi.org/10.1007/978-3-030-58607-2_41.

[27] Le, H. T., Phung, S. L., & Bouzerdoum, A. (2021). A fast and compact deep Gabor network for micro-Doppler signal processing and human motion classification. IEEE Sensors Journal, 21(20), 23085–23097.

[28] Liu, H., Ruan, Z., Zhao, P., Dong, C., Shang, F., Liu, Y., Yang, L., & Timofte, R. (2022). Video super-resolution based on deep learning: A comprehensive survey. Artificial Intelligence Review, 55(8), 5981–6035.

[29] Liu, X., Shi, K., Wang, Z., & Chen, J. (2021). Exploit camera raw data for video super-resolution via hidden Markov model inference. IEEE Transactions on Image Processing, 30, 2127–2140.

[30] Ma, J., Yu, J., Liu, S., Chen, L., Li, X., Feng, J., Chen, Z., Zeng, S., Liu, X., & Cheng, S. (2020). PathSRGAN: multi-supervised super-resolution for cytopathological images using generative adversarial network. IEEE Transactions on Medical Imaging, 39(9), 2920–2930.

[31] Maggiori, E., Tarabalka, Y., Charpiat, G., & Alliez, P. (2017). High-resolution aerial image labeling with convolutional neural networks. IEEE Transactions on Geoscience and Remote Sensing, 55(12), 7092–7103.

[32] Mahmoud, M., & Kang, H.-S. (2023). GANMasker: A Two-Stage Generative Adversarial Network for High-Quality Face Mask Removal. Sensors, 23(16), 7094.

[33] Mungoli, N. (2023). Scalable, Distributed AI Frameworks: Leveraging Cloud Computing for Enhanced Deep Learning Performance and Efficiency. ArXiv Preprint ArXiv:2304.13738.

[34] Neudorfer, C., Butenko, K., Oxenford, S., Rajamani, N., Achtzehn, J., Goede, L., Hollunder, B., Ríos, A. S., Hart, L., & Tasserie, J. (2023). Lead-DBS v3. 0: Mapping deep brain stimulation effects to local anatomy and global networks. Neuroimage, 268, 119862.

[35] Singh, A., & Singh, J. (2019). Review and comparative analysis of various image interpolation techniques. 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), 1, 1214–1218.

[36] Sun, N., & Li, H. (2019). Super resolution reconstruction of images based on interpolation and full convolutional neural network and application in medical fields. IEEE Access, 7, 186470–186479.

[37] Peng, D., Bruzzone, L., Zhang, Y., Guan, H., Ding, H., & Huang, X. (2020). SemiCDNet: A semisupervised convolutional neural network for change detection in high resolution remote-sensing images. IEEE Transactions on Geoscience and Remote Sensing, 59(7), 5891-5906.

[38] Tang, G., Ni, J., Chen, Y., Cao, W., & Yang, S. X. (2023). An Improved CycleGAN Based Model For Low-light Image Enhancement. IEEE Sensors Journal.

[39] Huang, Y.-W. and Chen, J. (2022) *Improved EDVR model for robust and efficient video super-resolution*, *ieeexplore.ieee.org*. Available at: https://ieeexplore.ieee.org/document/9707569/ (Accessed: 10 July 2023).

[40] Abyaneh, A.Y., Foumani, A.H.G. and Pourahmadi, V. (2018) "Deep neural networks meet CSI-Based authentication.," *arXiv (Cornell University)* [Preprint]. Available at: http://export.arxiv.org/pdf/1812.04715.

[41] Lu, Z. *et al.* (2023) *Can sam boost video super-resolution?*, *arXiv.org*. Available at: https://arxiv.org/abs/2305.06524v2 (Accessed: 05 July 2023).

[42] Kirillov, A.M. *et al.* (2023b) "Segment anything," *arXiv (Cornell University)* [Preprint]. Available at: https://doi.org/10.48550/arxiv.2304.02643.

[43] Vlaović, J., Žagar, D., Rimac-Drlje, S., & Vranješ, M. (2021). Evaluation of objective video quality assessment methods on video sequences with different spatial and temporal activity encoded at different spatial resolutions. International Journal of Electrical and Computer Engineering Systems, 12(1), 1–9.

[44] Wang, Y., Wang, L., Yang, J., An, W., & Guo, Y. (2019). Flickr1024: A large-scale dataset for stereo image super-resolution. Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, 0–0.

[45] Wang, Z., Chen, J., & Hoi, S. C. (2020). Deep learning for image super-resolution: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(10), 3365–3387.

[46] Wei, J., Zhou, C., Wang, J., & Chen, Z. (2022). Time-Series FY4A Datasets for Super-Resolution Benchmarking of Meteorological Satellite Images. Remote Sensing, 14(21), 5594.

[47] Xiao, Z., Xiong, Z., Fu, X., Liu, D., & Zha, Z. J. (2020, October). Space-time video super-resolution using temporal profiles. In Proceedings of the 28th ACM International Conference on Multimedia (pp. 664-672).

[48] Yan, Q., Chen, W., Zhang, S., Zhu, Y., Sun, J., & Zhang, Y. (2023). A Unified HDR Imaging Method with Pixel and Patch Level. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 22211–22220.

[49] Yi, P., Wang, Z., Jiang, K., Jiang, J., Lu, T., & Ma, J. (2020). A progressive fusion generative adversarial network for realistic and consistent video super-resolution. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(5), 2264–2280.

[50] Zhang, S., Mao, W., & Wang, Z. (2023). An efficient accelerator based on lightweight deformable 3D-CNN for video super-resolution. IEEE Transactions on Circuits and Systems I: Regular Papers.

[51] Zhao, X., Ding, W., An, Y., Du, Y., Yu, T., Li, M., Tang, M., & Wang, J. (2023). Fast Segment Anything. ArXiv Preprint ArXiv:2306.12156.

[52] Zeng, Y., Liu, X., Xiao, N., Li, Y., Jiang, Y., Feng, J., & Guo, S. (2019). Automatic diagnosis based on spatial information fusion feature for intracranial aneurysm. IEEE transactions on medical imaging, 39(5), 1448-1458.

[53] Bevilacqua, M., Roumy, A., Guillemot, C., & Alberi-Morel, M. L. (2012). Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In Proceedings of the British Machine Vision Conference (pp. 1-10).

[54] Zeyde, R., Elad, M., & Protter, M. (2010). On single image scale-up using sparse-representations. In International conference on curves and surfaces (pp. 711-730). Springer, Berlin, Heidelberg.

[55] Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001). 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 (Cat. No. 01CH37205) (Vol. 2, pp. 416-423). IEEE.

[56] Huang, J.-B., Singh, A., Ahuja, N., & Yang, M.-H. (2015). Single image super-resolution from transformed self-exemplars. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 5197-5206).

[57] Agustsson, E., & Timofte, R. (2017). NTIRE 2017 challenge on single image super-resolution: Dataset and study. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 1110-1121).

[58] Matsui, Y., Hamamoto, T., & Kato, K. (2017). Sketch-based manga retrieval using manga109 dataset with annotation of mangaka styles. In Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval (pp. 399-402).

[59] Liu, Z., Luo, P., Wang, X., & Tang, X. (2015). Deep learning face attributes in the wild. In Proceedings of the IEEE International Conference on Computer Vision (pp. 3730-3738).

[60] , B., Son, S., Kim, H., Nah, S., & Lee, K.M. (2017). Enhanced deep residual networks for single image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 136-144).

[61] *NTIRE Workshop*, *NTIRE2017: New trends in image restoration and enhancement workshop and challenge on Image Super-resolution*. Available at: http://www.vision.ee.ethz.ch/ntire17/ (Accessed: 10 August 2023).

[62] PIRM Challenge: https://www.pirm2018.org/

[63] PIRM Challenge , Leadership 2018-2020 - workshops, content and more. Available at: https://www.pirm2018.org/ (Accessed: 10 August 2023).

[64] Nah, S., Kim, T. H., & Lee, K. M. (2019). Deep multi-scale convolutional neural network for dynamic scene deblurring. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 257-266)

[65] Wang, X., Chang, K., *et al.* (2019) *EDVR: Video Restoration with Enhanced Deformable Convolutional Networks*. Available at: https://xinntao.github.io/projects/EDVR (Accessed: June 2023).

[66] Xinntao (no date) *Xinntao/EDVR: EDVR has been merged into BASICSR.*, *GitHub*. Available at: https://github.com/xinntao/EDVR (Accessed: 10 July 2023).

[67] Sun, X. *et al.* (2021) "VSRNet: End-to-end video segment retrieval with text query," *Pattern Recognition*, 119, p. 108027. Available at: https://doi.org/10.1016/j.patcog.2021.108027.

[68] Chan, K.C.K. (2020) *BasicVSR: The search for essential components in Video Super-Resolution and Beyond*. Available at: https://arxiv.org/abs/2012.02181

[69] Kim, J., Lee, J.K. and Lee, K.M. (2015) "Accurate Image Super-Resolution Using Very Deep Convolutional Networks," *arXiv (Cornell University)* [Preprint]. Available at: https://doi.org/10.48550/arxiv.1511.04587.

[70] Tan, M. and Le, Q.V. (2019) “EfficientNet: Rethinking model scaling for convolutional neural networks,” *International Conference on Machine Learning*, pp. 6105–6114. Available at: http://proceedings.mlr.press/v97/tan19a/tan19a.pdf.

[71] He, K. *et al.* (2015) “Deep Residual Learning for Image Recognition,” *arXiv (Cornell University)* [Preprint]. Available at: https://doi.org/10.48550/arxiv.1512.03385.

[72] Wang, X. *et al.* (2022) “A lightweight modified YOLOX network using coordinate attention mechanism for PCB surface defect detection,” *IEEE Sensors Journal*, 22(21), pp. 20910–20920. Available at: https://doi.org/10.1109/jsen.2022.3208580

[73] Bochkovskiy, A. (2020) *YOLOV4: Optimal speed and accuracy of object detection*. Available at: https://arxiv.org/abs/2004.10934v1.

[74] Terven, J.R. and Cordova-Esparza, D. (2023b) “A comprehensive review of YOLO: from YOLOV1 and beyond,” *arXiv (Cornell University)* [Preprint]. Available at: https://doi.org/10.48550/arxiv.2304.00501.