EMODnet Thematic Lot n°1 – Bathymetry

CINEA/EMFAF/2022-23/3.5.2/01/SI2.927891

Start date of the project: 20/12/2024 (24 months)

Operational Phase

D4.4 - Interim report on AI technology exploration

November 2025

The European Marine Observation and Data Network (EMODnet) is financed by the European Union under Regulation (EU) 2021/1139 of the European Parliament and of the Council of 7 July 2021 establishing the European Maritime, Fisheries and Aquaculture Fund.

Disclaimer

Document info

Title (and reference)	Interim report on AI technology exploration (D4.4)
WP title (and reference number)	Technical Development & Operation of services and tools (WP4)
Task (and reference number)	Prototype development of AI technology for EMODnet Bathymetry (WP4.4)
Authors [affiliation]	Ricard Campos (Coronis), Willem Tromp (Deltares), Martin Verlaan (Deltares), Sina Starmans (EOMAP)
Dissemination level	Public
Submission date	19-12-2025
Deliverable due date	14-11-2025

Contents

1 Introduction

Every 2 years EMODnet releases a gridded bathymetry for all European waters. This is a core resource for numerous studies, as can be seen from the large number of downloads, for example. The development of this gridded bathymetry from the collected source surveys is a complex process that uses a large number of datasets and contains many steps. Some of the processing steps are laborious and can potentially be assisted by Artificial Intelligence (AI). Moreover, AI can potentially also draw in other types of data, and can be used to create improved or completely new products.

This report provides a literature review of Artificial Intelligence (AI) techniques that are or can potentially be applied to the processing of bathymetric datasets. The focus is on the work as carried out for EMODnet bathymetry, but may be applicable more widely. The reader will notice that many methods originate from other fields, such as image processing, that lead the way for many other application domains. These image processing algorithms can sometimes be applied directly, by interpreting bathymetry as a scaled version of a gray-scale image. But more often, the methods from this field serve as inspiration to develop more specialized methods that better consider the specific properties of bathymetry.

In this report, first an overview is given of the processing steps used to create the EMODnet gridded bathymetry and several related products. Next, the bulk of the report will focus on three potential applications of AI in this process. Finally, the logical next steps and challenges are discussed.

1.1 The bathymetry processing workflow in EMODnet

EMODnet provides a wide range of bathymetry products. Arguably, the most well-known product is the EMODnet gridded bathymetry, which is created from thousands of datasets contributed by dozens of institutes across Europe and serves as a core resource for many users. The process is similar to the workflow described by (Jacobsson et al 2012). The figure below gives an overview of the steps and datasets involved. The main steps after collection of the datasets are the stage 1 processing that results in a basic gridded bathymetry per region. In this context the regions are a subdivision of the European waters in order to divide the work over a number of basin coordinators, with this also benefitting from the regional expertise of the coordinator and their network. Next, the stage 2 processing combines the regional gridded bathymetries, performs additional checks and complements the data. Finally, different products are generated from the raw products. For example downloads in various formats, correction from LAT to MSL, a dataset that can be used for 3D rendering, various maps, etc. The next sections describe these steps in more detail and highlight some of the opportunities for application of AI.

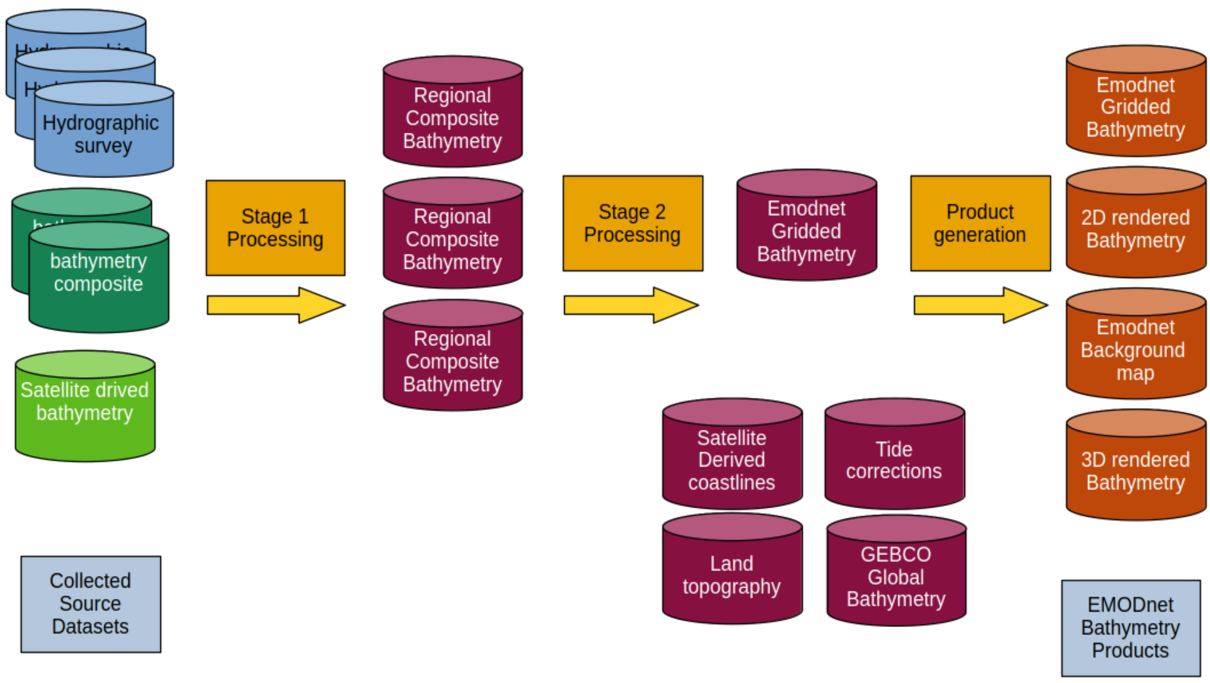


Figure 1: EMODnet workflow for bathymetry.

1.2 Stage 1 processing

Every cycle of EMODnet bathymetry starts with updating and collecting new hydrographic surveys and adding them all to a catalogue. In several coastal areas also Satellite Derived Bathymetry (SDB) is available. In total, the catalogue contains many thousands of datasets. The first stage processing aims to compile regional composite gridded bathymetries. In this stage, the following steps are needed:

- **Conversion:** The datasets collected may use different file formats, use different coordinate systems. All datasets are converted to a common format to allow for efficient direct comparison.
- **Determine priority:** Surveys may overlap, and merging the datasets requires choices on which datasets should get priority over others. An example is where later surveys get priority over older ones. This step also includes an assessment of the quality of each dataset. EMODnet uses a standardised set of metadata attributes that can assist the process, but this also requires expert judgement.
- **Flag bad data**: Datasets can contain outliers or other artefacts that need removal. Several tools in the Globe software used in EMODnet are in use to assist this step. Still, this step is currently in part manual.
- **Fill small data voids**: Some data points may be removed as a result of quality flags during the acquisition, but also in the previous step. Often, it is undesirable that small holes in the dataset are filled with values from a dataset with lower priority, since this usually gives poorer results than interpolation/gap filling/inpainting. For somewhat larger gaps, datasets with lower priority do provide valuable input.

- **Data merging**: Where data from different surveys are next to each other, artefacts may appear, and some smoothing/correction may be required. If there are clear quality differences between the datasets, then the corrections should mostly be applied to the dataset of lower priority.
- **Bias correction**: Some datasets may appear shifted compared to other datasets. Minor differences are expected and result from surveys performed at different times, under different conditions and with different corrections applied to them. Sometimes, larger differences can result from mistakes, such as not correcting for the variability of the speeds of sound in seawater. Also other systematic artefacts may appear, for example caused by a combination of coordinate transformation and interpolation.

Only a few publications were found that explicitly addressed the issues above. For example Shar (Shaw, 1993) applied an automated procedure to look for errors in the dataset, but the approach is more classical statistical than AI based. However, there is much AI literature for image processing. The task of flagging bad data can be seen as image segmentation. Image techniques can often not be applied in their present form, but can serve as a starting point for the development of similar techniques in the bathymetry field. Many of the tasks can potentially be assisted by AI, but in this report the focus is on the detection and correction of errors (section 4).

1.3 Stage 2 processing

In the second stage, the regional bathymetries are combined into a single Europe wide gridded bathymetry. This requires several steps:

- **Merge regions**: First the regional datasets are merged into a single dataset. The regional bathymetries overlap a bit at the edges to support the integration. Differences between regions are discussed between coordinators, which can lead to corrections, but smaller differences are smoothed in the overlap.
- **Add coastline**: The bathymetry in the cells closest to the coast are aligned with the coastline. The coastline is a separate product that is derived from several sources, such as Open Streetmap and Satellite imagery.
- **Fill gaps**: Areas where no surveys are available are filled with data from other datasets or interpolated. One dataset used in some parts of the bathymetry is GEBCO. In some places filling with GEBCO data may result in large artefacts, and other datasets or methods are used.
- **Add land topography**: For some applications it's useful to have a dataset that covers both sea and land, so the bathymetry is complemented with topographic data from various sources. At the coastline, these should connect.
- **Final review**: At this point the raw gridded bathymetry is nearly complete. The data undergoes a final check before it's released to generate the products.

Filling gaps is especially complicated near the coast, since interpolation generally ignores consistency with the coastline. In addition, bathymetry is often missing relatively often near the coast, because different institutes are responsible for surveying. This combination can lead to significant artefacts in filling the bathymetry there. AI can potentially assist to generate more accurate and/or realistic bathymetry near the coast when surveys are lacking. Section 3 addresses this issue.

1.4 Product generation

The combined gridded bathymetry that results from the stage 2 processing serves as the main input for generating the various products. Without trying to be complete some products are:

- EMODnet gridded bathymetry tiles in various formats as offered for download
- EMODnet gridded bathymetry corrected to MSL (instead of LAT)

- Bathymetry maps, i.e. image tiles that can be served for example as the background maps in the EMODnet map portal.
- Dataset that can be served for 3D rendering
- High-resolution bathymetry for selected regions.

For some of the products the accuracy is key, but for rendering the results, more weight could be assigned to the aesthetics of the results. This is one area where AI super-resolution could shine. Section 2 discusses this topic.

2 AI Super-Resolution

Gathering high-resolution bathymetry is crucial for understanding seafloor relief. However, obtaining bathymetric maps at a decent resolution is both operationally difficult and costly. While multi-beam sonar technologies may allow to collect dense data and compose high-resolution maps, depending on the area these maps are collected at a high altitude. This results in many sources for bathymetry being of lower resolution compared to land data. For these reasons, one of the well-known topics in Deep Learning that is identified as being of relevance for EMODnet Bathymetry is super-resolution (SR).

Also known as image upsampling, SR aims at creating a high-resolution (HR) image from a low-resolution (LR) version. If applied to Bathymetry, this would translate into obtaining maps with higher resolution from low resolution sources. Given the complexity of gathering new samples on certain areas, having a proper way to increment the resolution of some bathymetric maps in a realistic fashion is much desirable in the bathymetric domain.

Deep-learning (DL) SR has been a hot topic in recent years, as it can be applied to a wide variety of real-world applications such as streaming services (by transferring LR images that are lighter in terms of size that are then upsampled), surveillance (to improve security cameras footage), remote sensing (to improve satellite images), medical imaging (to enhance MRI and/or CT scans), etc. Basically, all image-related methods may benefit from having a higher-resolution version of an image, provided that the upsampling is both aesthetic and plausible.

In this section, the state of the art is reviewed in SR techniques and their possible implementation for bathymetric data. More precisely, the next section starts with defining the problem and the common metrics used to quantify the correctness of a SR result. Once these base concepts are defined, a comprehensive review is provided of DL SR methods in the computer vision/image processing community, dividing them into classical and DL approaches. Next, the state of the art is examined in DL SR applied specifically to the bathymetry topic. Finally, some conclusions are drawn and the next steps are foreseen to implement SR methods that are useful for the EMODnet bathymetry community.

2.1 Problem Definition

As mentioned above, the SR problem aims at generating a HR version of a LR raster/image. It is important to start by noting that SR is an ill-posed problem: given an LR raster, there are always multiple possible HR images that may correspond to such LR versions. This stems from the fact that going from a HR raster to its LR counterpart is a *degradation* process, as shown in this equation:

$$I_L = D(I_H, \sigma) \quad (1)$$

Where I_L is the low-resolution image, I_H is the high-resolution, and D is the degradation function, σ denoting its noise parameters. Usually, D and σ are unknown, and D may include operations such as blurring, downsampling or compression, among others.

Therefore, SR methods try to invert this process, and mainly increase the resolution of the image while recovering the details lost by the degradation function.

2.1.1 Metrics

Metrics provide a quantitative measure for how good the upscaled image is as compared to a reference *ground truth*, and may serve two purposes in this field. On the one hand, they allow quantification of the accuracy or correctness of a SR method and, on the other hand, they can be used to guide the minimization process during the optimization of a DL model.

The most common metrics are pixel-based, in the sense that they compute a measure based on a pixel-per-pixel difference between the ground truth and the reconstructed HR image. From this type of metrics, the Mean Squared Error (MSE) is the most frequently used, its definition being:

$$MSE = \frac{1}{n}\sum_{i=1}^{n} \quad (x_i - \widehat{x_i})^2$$

Where x_i and $\widehat{x_i}$ are the real and predicted value respectively. While some variants of this metric can also be used in the literature, such as the Mean Absolute Error (same equation without the squared operator) or the Root-Mean Squared error (same as MSE but with a square root applied), the MSE is the prevalent metric to minimize during training of neural networks. This is due to its differentiability, which makes it well-behaved for the type of gradient-based optimization used during deep neural networks' training.

However, in order to discern if a method is better than the other, objective metrics are needed that allow us to compare them. One of the most used metrics for performance comparison is the Peak Signal-to-Noise Ratio (PSNR):

$$PSNR = 10log_{10}\left(\frac{MAX^2}{MSE}\right)$$

Where MAX is the maximum pixel value of the image (255 for an 8-bit image). The result is in decibels (dB) and the higher its value the better the reconstruction quality as compared to the ground truth.

As can be seen from its equation, PSNR is directly related to the MSE. Therefore, both are based on computing differences at pixel level. However, the human vision system is keener to focusing on overall structural information when comparing two images. In this direction, instead of measuring pixel-by-pixel differences, the Structural Similarity Index (SSIM) compares luminance, contrast and structure between the images. Given two windows of pixels x and y, the SSIM is defined as:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Where μ_x, μ_y and σ_x^2, σ_y^2 are the pixels' mean and variance values of x and y windows, σ_{xy} their covariance, and c_1, c_2 small constants to avoid division by zero. An image with high MSE might still look structurally similar to the original, but since SSIM is more sensitive to structural features it would capture such similarity better.

While SSIM tries to improve on getting a measure that really represents a human observer, it is still based on simple statistics computed over pixel intensities. Therefore, it cannot capture high-level perceptual or semantic differences that humans normally focus on. Consequently, the apparently ideal metric for measuring the behavior of a SR method would be to let human observers rate the quality of the results. The Mean Opinion Score (MOS) goes in this direction by letting a set of independent human reviewers rate the results subjectively (in a scale from 0 to 10, for instance), to then compute the mean of such answers.

However, while much desirable, collecting MOS measurements is resource and time consuming. Because of this, there are some proposals in the literature for substituting human intervention by some DL models that proved useful in detecting structural features in the images. An example of such metrics is the Learned Perceptual Image Patch Similarity (LPIPS), which uses some pre-trained deep networks for generating feature/activation maps from the ground truth and the reconstructed HR image to then compare them using the L2 distance weighted by learned coefficients that reflect perceptual relevance. In this way, the pre-trained network computes the relevant structural details from the ground truth HR image and compares them to the ones generated by the reconstructed one (ideally, they should match). In the same direction, the Fréchet Inception Distance (FID) does a similar comparison, but at dataset level instead of using only a single-image, by comparing the mean and covariance of deep features extracted using the Inception-v3 network via Fréchet distance:

$$FID = \|\mu_r - \mu_g\|^2 + Tr\left(\sigma_r + \sigma_g - 2(\sigma_r\sigma_g)^{1/2}\right)$$

In this equation each feature set is modelled as a multi-variate Gaussians with mean μ_r, μ_g and covariance σ_r, σ_g for the real and the ground truth datasets respectively, and $Tr()$ is the *trace* operator.

Overall, the take-home message from this section is that there is no gold standard for measuring the quality of a SR result, so it is important to know what the main metrics presented are contrasting when reviewing the results for the different methods in the state of the art. For instance, a high PSNR does not always mean a better reconstruction perceptually, but it is found to be normally used when comparing results across methods. Also, the use of metrics based on deep-learning features may be opinionated, since such feature detectors may be trained with datasets that may or may not resemble the features being reconstructed by the SR methods.

2.2 Classical Super-Resolution Methods

One cannot dive into the recent SR methods without first delving into the classical approaches. According to many reviews on the topic (Al-Mekhlafi and Liu, 2023; Huang and Liu, 2015), classical SR methods prior to the DL era can be broadly classified into being *interpolation-based* or *reconstruction-based* approaches.

On the one hand, interpolation-based methods just upsample the image grid resolution and use an interpolation method to fill the gaps between known data points (see Figure 2). The most common ones are *nearest-neighbor*, *bilinear* and *bicubic* interpolation.

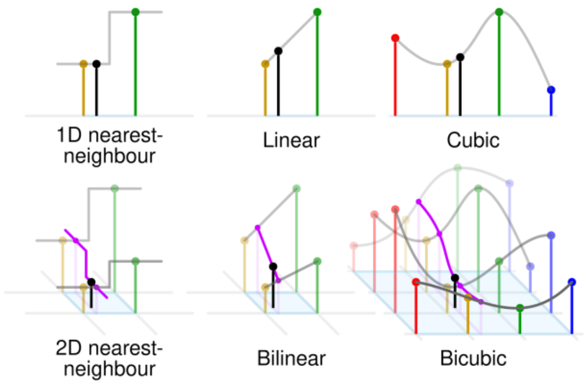


Figure 2. Classic interpolation methods in 1D and 2D, the black dot represents the interpolated point. Image by Cmglee - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=53064904

The advantages of such methods are basically their ease of implementation and their very low computational cost. However, their main problem is that, while they work correctly in smooth areas of the images, they may blur sharp edges, create blocky artifacts, and cannot generate high frequency components (i.e., they cannot recover fine details) (van Ouwerkerk, 2006).

Note that interpolation-based methods are just treating SR as a gap-filling problem, and do not follow the SR definition given in the previous section. On the other hand, reconstruction-based methods do consider that the problem basically resides in inverting the degradation function D shown in equation (1) and try to recover such a degradation process. The most relevant methods following this methodology are Iterative Back-Projection (IBP), Projection Onto Convex Sets (POCS) (Stark and Oskoui, 1989) and Maximum a Posteriori (MAP) (Schultz and Stevenson, 1996, 1995). However, these methods require not just a single LR image as input but a *sequence* of LR images that allows recovering high-frequency information by combining all the observations. Therefore, while they may be useful for multi-images or video feeds, they are not useful in the bathymetry domain since there is no such redundancy available for LR bathymetric maps.

2.3 Deep Learning Super-Resolution Methods

One could see the methods in the previous section as *hand-crafted*. That is, they try to define the inverted degradation process as an algorithm. On the contrary, DL methods take sample pairs of LR-HR images and train on these inputs a given statistical model that learns how to output a HR image from a LR input.

Due to the intended use of the methods for bathymetric SR, the scope of this review does NOT include the following topics:

- Multi-image: as mentioned in the previous section, the most advanced classical methods prior to the use of DL rely on having multiple LR images, since mixing information from a sequence of images allows extracting more information for the HR reconstruction. However, in bathymetry it is rarely the case to have the same area at multiple resolutions (at least the same full image). Therefore, in this review the methods that rely on multiple frames are not taken into account.
- Real-time application: many SR methods in the literature strive at attaining real-time applications to ingest video-input that can be processed on the go. Since bathymetric SR is not constrained (a priori) by the real-time factor, those methods that modify or add special tricks to some of the models just to speed up the process (usually at the cost of attaining slightly worse results) are not considered.

Once the scope of this review has been defined, a chronological listing of the DL SR methods in the state of the art will be given, in which it will be tried to summarize the contributions of each article/model with respect to their predecessors in terms of their network architecture. In this sense, main architectures used is broadly divided into:

- Convolutional Neural Networks (CNNs).
- Generative Adversarial Networks (GANs).
- Denoising Diffusion Probabilistic Models (DM).
- Image Transformers (IT).

This classification coincides chronologically with the main trends followed by the image processing DL community through the years, but keep in mind that it is not unique, since some architectures may be complex enough as to include parts from other architectures. For instance, many GANs, DMs, and ITs use CNNs for some of their steps. However, while this classification may be opinionated at some points, it is tried to focus on the main architecture and training procedure ruling a method for providing the proposed separation.

2.3.1 Convolutional Neural Networks

A Convolutional Neural Network (CNN) is a deep learning model specifically designed to work with gridded data such as images/rasters. Unlike traditional neural networks, CNNs do not treat all input features equally, since they use specialized layers named *convolutional layers* that apply learnable filters across the input. Such filters detect/capture local patterns such as edges, textures, shapes, etc. The filters are convolved across the image, preserving spatial relationships between pixels, which makes CNNs particularly effective at recognizing visual patterns regardless of where they appear in the image.

One of the first methods to deal with SR as a Deep Learning problem is SRCNN (Dong et al., 2016a) which, as its name suggests, presents a typical CNN architecture. This method starts by upscaling the image using a classical method (bilinear/bicubic) to then apply a CNN to correct this initial approximation for the upsampling to get better details and sharpness. The CNN in SRCNN, shown in Figure 3, consists of three layers: the first is a patch extraction layer, which extracts patches from the upsampled input and represents them using convolutional filters. Then, the second is a non-linear mapping layer consisting of 1x1 convolution filters that change the number of channels and adds non-linearity. Finally, the third is the reconstruction layer, responsible for composing the high-resolution image.

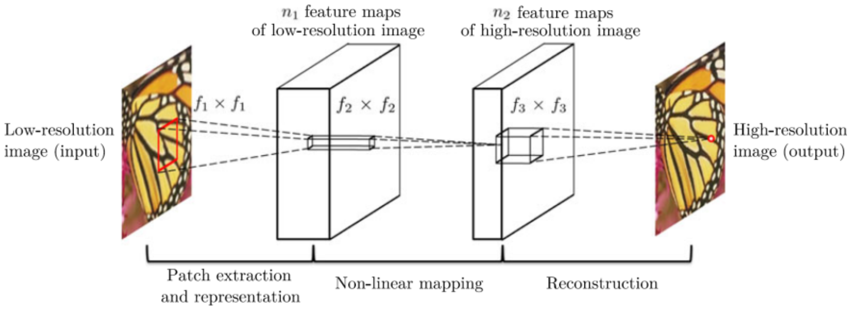
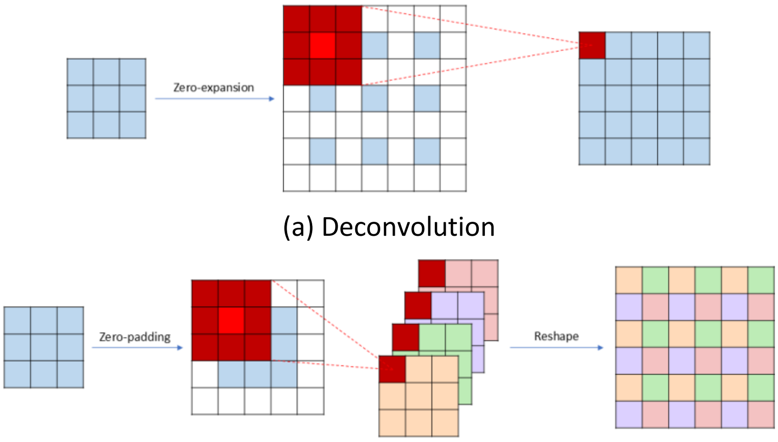


Figure 3. SRCNN architecture. Figure extracted from (Dong et al., 2016a).

The simplicity of this network made it popular due to its ease of implementation on modern Deep Learning frameworks, as it is based on simple (though large) convolutional layers. In fact, if one had no restrictions in the depth of the convolutional layers one can attain in terms of number of parameters, and there existed an ideal optimization method able to tune the network regardless of its size with an acceptable computational cost, this simple structure could obtain far better results than those presented in the original reference. Since such ideal conditions do not apply in the real world, the following sub-sections present the different approaches proposed in the literature to improve CNN-based models in terms of their size and/or training complexity.

2.3.1.1 Upsampling strategy

Before delving into the main strategies for overcoming the limitations of using a naïve CNN architecture as the one presented in SRCNN, it should be noticed that CNN-based SR methods can work in two different ways with respect to when the *upsampling* takes place. For instance, one has seen in the previous section that in SRCNN (Dong et al., 2016a) the image is upsampled as a first step using a classical method (usually bicubic interpolation), and then the CNN is left to recover the missing *details*. Methods following this approach promote the networks to be larger in the first layers, as they need to extract features in the HR space. To reduce the size of the network in these first steps, methods exploring the possibility of upsampling *at the end* of the network started appearing. Also, by doing so, the upsampling process is also part of the learned network, and does not follow a pre-defined methodology. There are mainly two different proposals in the literature for how to perform this learnable upsampling step able to increase the spatial resolution of a feature map in a CNN. The first proposal is the *deconvolution filter* (also referred to in some articles as the *transposed convolution*), which was first presented as part of the FSRCNN model (Dong et al., 2016b). As its name suggests, the deconvolution filter is a mathematical operation that reverses the convolution operator. As shown in Figure 4 (a), it first upsamples the layer by adding zeros in-between the original pixels/cells in a process called *zero-expansion*, and then applies a convolution operator to get the final value. Since the original layer was upscaled using the zero-expansion, so is the convolved result. By using this trick, FSRCNN achieves better results than SRCNN while also attaining a faster performance. The second upsampling proposal was the *subpixel convolution* introduced in ESPCN (Shi et al., 2016) (also referred to as *pixel-shuffle* in some references) which, as shown in see Figure 4 (b), instead of learning the direct upsampling, learns how to produce an interleaved feature map which is then rearranged into a HR grid. From these two, subpixel convolution avoids some blocky artifacts produced by the deconvolution filter while also being more efficient (Odena et al., 2016).



(a) Deconvolution

(b) Sub-pixel convolution

Figure 4. Deconvolution (a) and Sub-pixel convolution (b) upsampling operators. The red cells/lines indicate the common convolution operator, while white cells represent zeros.

2.3.1.2 Residual Networks

During training of a neural network, information flows in two directions:

- The *forward pass*, where the input data is passed through the network, being transformed layer by layer into high-level features and finally predictions.
- The *backward pass*, once the forward pass finishes, the prediction is compared with some reference data and the errors (gradients) are propagated backward through the network's layers to update their weights.

Obviously, when the network is very deep, this optimization may become unstable or directly break down, leading to the so-called *information flow* issues, such as vanishing/exploding gradients (Pascanu et al., 2013) or the degradation problem (He et al., 2016).

The residual network architecture (ResNet) explicitly tackles this problem by adding *skip connections*, which allow signals to flow more directly across layers, effectively solving a lot of the optimization difficulties of deep nets related to the information flow.

In fact, one of the first improvements over the initial pure CNN approximation proposed by SRCNN was presented in the VDSR model (Kim et al., 2016a), where instead of a small architecture with large convolutional filters, the authors resorted to a deeper network with smaller convolutional filters of size 3x3 and, in addition, they use a Residual Network approach. As shown in Figure 5, instead of trying to directly learn the mapping from LR to HR, the network learns the *residual* between the naively-upscaled input and the final HR image. This greatly reduces the complexity of the learning task, as the model does not learn *how to reconstruct the whole image* but just the *missing details*.

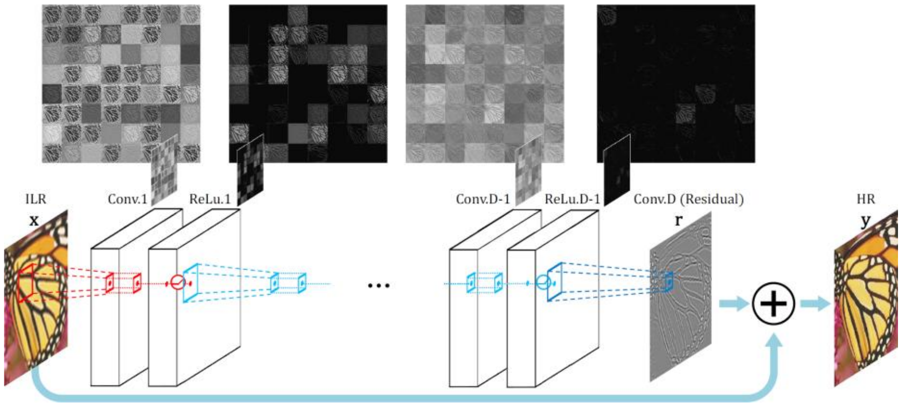


Figure 5. VDSR architecture. Figure extracted from (Kim et al., 2016a).

The improvement of using a residual network seen in VDSR can also be mixed with performing the upsampling at the end of the network. Samples of these are for instance SRResNet (Ledig et al., 2017) and EDSR (Lim et al., 2017), where the main difference between them is that EDSR removes the batch normalization layers used in SRResNet. The authors of EDSR state that Batch Normalization layers limit the range of the network and thus lead to worse results while, at the same time, taking more memory. Therefore, removing them increases both computational cost and performance.

Continuing with residual networks, CARN (Ahn et al., 2018) introduces a cascading mechanism to incorporate features from multiple layers and allow the network to receive more information from other parts, not just from the prior CNN step/block.

All the methods presented so far have the upsampling step either at the beginning or at the end of the network. Contrarily, BTSRN (Fan et al., 2017) proposes to upscale at the middle, leading like this to a *multi-stage* residual network, where they separate the model into the LR stage and the HR stage. The LR stage consists of 6 residual blocks, and its output is upsampled using a deconvolution layer and nearest neighbor upsampling, to then enter the HR stage, composed of four residual blocks working in the HR space. Also, the residual block they propose differs a bit from its predecessors.

2.3.1.3 Recursive Networks

While residual networks tackle the problem of training complexity, the networks still grow larger the deeper they are in terms of number of parameters to optimize. To deal with this problem, Recursive Networks use the same convolutional layers many times (hence the *recursivity*). Consequently, the number of parameters is greatly reduced, as the parameters of these recursively used convolutional layers are *shared*. Following this strategy, we can find DRCN (Kim et al., 2016b), where a single convolutional layer is applied recursively many times. This recursion allows the network to capture a very large receptive field (i.e., the region of the input image that affects a particular neuron output) and long-range dependencies without extra parameters. At the end of the network there is a reconstruction layer mapping all the final feature maps back to the HR image. In this last phase, since multiple inputs generate a single output, this architecture can be considered an *ensemble of* networks. The CARN-M variant presented in the original CARN paper (Ahn et al., 2018) also suggests that sharing parameters improved memory footprint while attaining comparable correctness. Moreover, the DRRN (Tai et al., 2017) proposal improves over DRCN by substituting the convolutional layers by full residual blocks (the main network block used in SRResNet (Ledig et al., 2017)). As DRCN did with the convolutional layers, the parameters were shared across different iterations of the same residual block. A comparison between two pure residual networks (SRResNet and VDSR) and two also recursive networks (DRCN and DRRN) is shown in Figure 6.

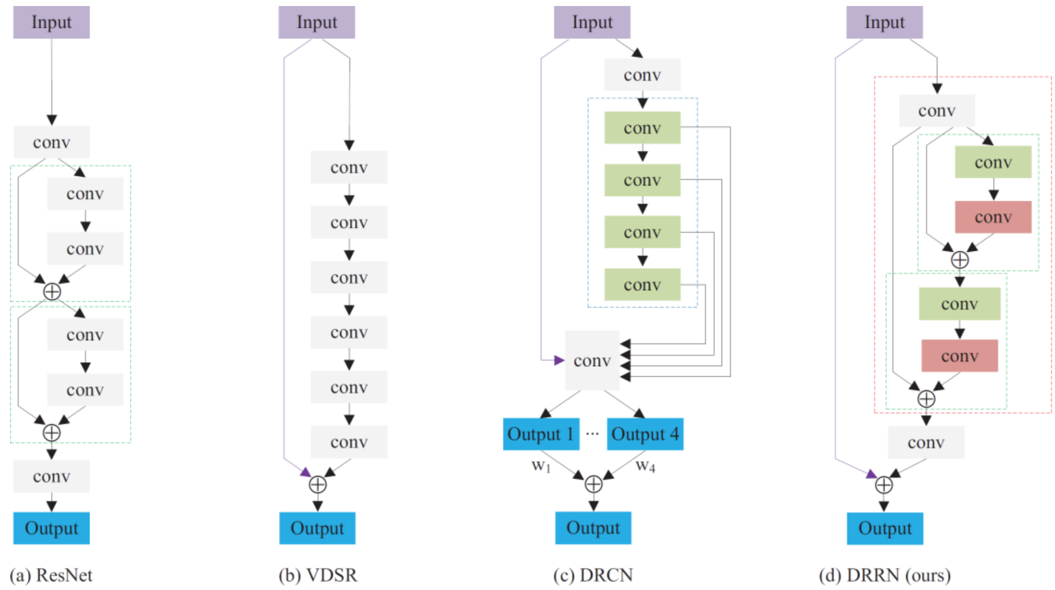


Figure 6. Architecture comparison between SRResNet (Ledig et al., 2017), VDSR (Kim et al., 2016a), DRCN (Kim et al., 2016b) and DRRN (Tai et al., 2017) models. Figure extracted from (Tai et al., 2017).

2.3.1.4 Progressive Networks

A common drawback of all the methods reviewed so far is that the upscaling factor of all the models is fixed. That is, a DL network is trained to perform HR at a single scaling factor of x2, or x3, or x4, etc. Therefore, individual models should be trained for each scaling factor desired. Trying to deal with this problem, in the same article of EDSR (Lim et al., 2017) they propose MDSR, an extension of the method allowing for multiple upscaling factors within a single model. They achieve this by making the first part of the model shared across scales, to then create different scale-specific upsampling layers at the end of the network.

Instead of creating different upsampling layers at the end, progressive networks increase the resolution of images in different steps, so that the upscaling of one block of the network is also the input for the next, doing a progressive upscaling at each step. The LAPSRN and MS-LAPSRN models (Lai et al., 2017) add the possibility of achieving multiple scaling factors, as EDSR, but using such progressive structure (and recursive networks) in this case. Based on the well-known Laplacian pyramid structure (Burt and Adelson, 1983), Figure 7 shows its structure as compared to pre/post upscaling networks dealing with a single scaling factor. The iterative building block of their network proposal consists of a feature embedding block followed by a feature upsampling block, upsampled via transposed convolution. This block learns a residual image that is added to a (bilinearly) interpolated upsampled image to get the next scale. Upon getting with this block, for instance, a x2 factor, the block for getting the x4 factor starts with the x2 as input and replicates the same network block, sharing the parameters across the different scales. The authors claim that using the low-resolution outputs to refine further scaling factors via a shared learning process attains better performance. They also noticed that, since a LR image can have multiple HR representations, the L2 loss function tends to generate overly smoothed results. For this reason, they propose to use the Charbonnier loss instead.

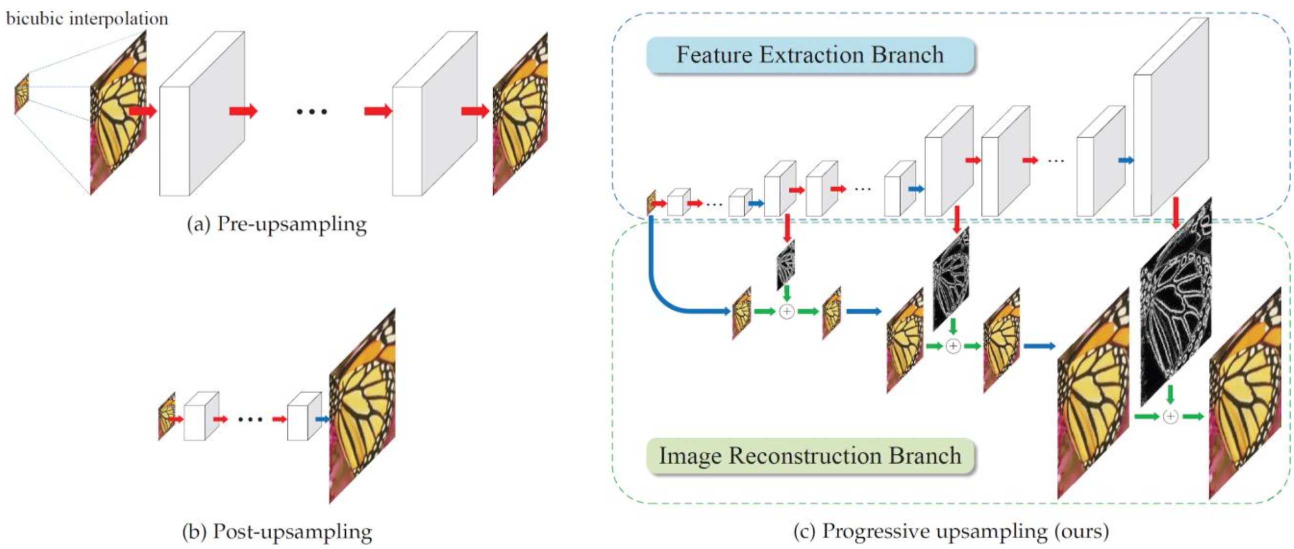


Figure 7. The idea of progressive network as compared to pre or post-upsampling networks. Figure extracted from (Lai et al., 2017).

2.3.1.5 Multi-branch Networks

As stated before, ResNets help mitigate the information flow problem via their skip connections. However, the fact that having an ever-deeper network improves results still applies. Therefore, as networks grow larger, the effectiveness of ResNets in solving information flow issues diminishes, complicating their optimization. In this sense, multi-branch networks use multiple branches, each focusing on different tasks/features, that allow the information to circulate through different paths, creating different receptive fields and improving information flow. Indeed, multiple paths in the network create shortcut connections, easing optimization and mitigating information flow issues (such as vanishing gradients) in deep networks.

The CMSC method (Hu et al., 2018) adds this multi-path concept to the idea of cascading residual networks introduced in CARN (Ahn et al., 2018). In this case, the cascaded sub-networks they use present two branches, each using different-sized filters (i.e., each branch focuses on different scales), which results in different receptive fields. The fusion of different receptive fields at the end of each block allows the information to flow better. Multiple blocks of this sub-network are stacked one after the other to gradually decrease the error between the output and the reference HR. The outputs of all the blocks are passed to a reconstruction block, responsible for composing the final output from the proposals of each subnetwork.

Another method using multiple branches to improve information flow is the Information Distillation Network (IDN) (Zong et al., 2022). Its key contribution is the Information Distillation Block (IDB), designed to progressively refine features while keeping the network compact. Instead of passing the extracted features through every layer (which causes redundancy), each IDB explicitly splits feature channels into two parts: a fraction of the most informative channels is kept for the final reconstruction (distilled features) and the rest are further processed in the block to refine them. Stacking several of these IDB blocks encourages the network to focus on important features first, while still later improving the weaker ones. Distilled features from all blocks are concatenated and fused at the end for the final SR reconstruction.

2.3.1.6 Attention Networks

The networks presented so far give equal relevance to all locations and channels of an image. Attention networks try to improve the results by selectively paying more attention to different regions of the image. This is achieved by reweighting feature responses, giving more importance to channels or spatial regions that are relevant for high-frequency texture reconstruction. This results in sharper edges, better texture recovery, and less over-smoothing.

In this direction, SelNet (Choi and Kim, 2017) introduces a selection unit after each convolutional block to determine which parts to selectively pass further down the network. This selection model consists of a ReLU activation function followed by a 1x1 convolution and a sigmoid gating mechanism. Similar to ESPCN, it uses a sub-pixel layer at the end of the network to achieve learned upscaling.

Another relevant contribution using attention networks is RCAN (Zhang et al., 2018), which adaptively scales channel-wise features based on their relevance. It introduces a hierarchical residual structure composed of a shallow feature extraction step, a Residual-in-Residual (RIR) network and an upsampling module. The RIR is their main contribution, and is composed of several blocks named Residual Group (RG), each of them containing another type of subblocks called Residual Channel Attention Blocks (RCAB). Each RG has a long skip connection and the entire RIR has also a skip connection around all RG. A visual depiction of such a complex structure is shown in Figure 8.

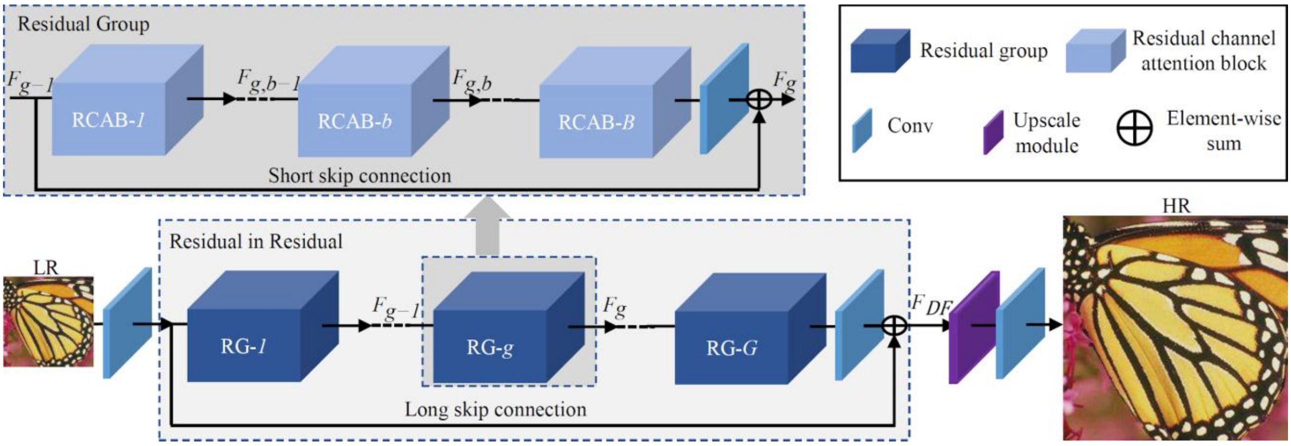


Figure 8. RCAN architecture. Figure extracted from the original RCAN reference (Zhang et al., 2018).

2.3.2 Generative Adversarial Networks

The error guiding the training of the networks described up to this point is based on comparing ground-truth HR images with the output of the network using some of the classical pixel-difference based metrics (MSE in most of the cases). Although this pixel-based metric works fine, it was already mentioned that it is not directly related to how humans perceive differences between two images (i.e., perceptual difference). Generative models (GANs) try to optimize for this perceptual quality to produce images that are recognized as realistic by a human observer. They achieve this by having two different networks. The first one is the *generator* network, which is mainly one of the networks discussed so far, trying to generate a HR image from a LR sample. The second one, named the *discriminator* network, tries to discern if an image is produced by an AI model or is on the contrary a real image. That is, it outputs a probability score measuring "how real" the input image looks. While both generator and discriminator networks are usually implemented via CNNs as those presented in the previous section, it is believed they represent a different type of DL architecture specially because of their unique training procedure that optimizes both networks at the same time. This optimization is called the *adversarial* process and consists in the generator trying to *fool* the discriminator by making more realistic images, while, at the same time, the discriminator is being trained to get better in discerning real from fake images. By doing this back-and-forth competition (represented by a minimax optimization during training), both networks improve their performance and the equilibrium is found when the generator produces images that are so realistic that the discriminator can no longer tell the difference (50% guess rate).

For instance, SRGAN (Ledig et al., 2017) optimizes the network to minimize three terms: (1) the typical MSE error, (2) the perceptual similarity loss, which uses another deep network to generate high-level features that are compared to the high-level features generated from the output image, and (3) an adversarial loss from a discriminator network. In EnhanceNet (Sajjadi et al., 2017) they also use a GAN structure, and add to the losses used in SRGAN a texture matching loss. As discovered in style transfer methods, the style of an image is captured by the Gram matrix (Horn and Johnson, 1985) of feature maps of a CNN (usually VGG), effectively capturing the texture statistics of the image. Therefore, the texture loss in EnhanceNet computes the difference between the Gram matrices of the real HR image and the inferred SR one. On its side, ESRGAN (Wang et al., 2019) improves the discriminator part of SRGAN by making it not only distinguish between fake/real images, but also to make real images look less real compared to the generated images. Their discriminator predicts relative realness, not absolute real/fake. This stabilizes the training of the network and encourages the generator to make fakes closer to reals, rather than just fooling the discriminator. Moreover, as suggested by other methods such as EDSR (Lim et al., 2017), they remove the Batch Normalization steps.

2.3.3 Denoising Diffusion Probabilistic Models

Also known as *diffusion models* (DM), this type of DL architectures generate data by reversing a gradual noising process. This noising process consists in progressively adding Gaussian noise to an image in a series of steps until the image is essentially indistinguishable from random noise. The model then learns how to *reverse* this process, therefore going from pure random noise to a realistic image. By breaking the generation into small denoising steps, each step is easier to learn by the network, resulting in more stable training and generation processes.

Initial approaches using this method were the SR3 (Saharia et al., 2023) and SRDiff (H. Li et al., 2022) methods, which are both based on the original Denoising Diffusion Probabilistic Models (DDPM) paper (Ho et al., 2020). Basically, in both SR3 and SRDiff methods the image is upscaled at the beginning using bilinear interpolation and the previously exposed diffusion/denoising process is used by progressively denoising a LR image into a HR one, conditioning every step on the given LR image. However, as shown in Figure 9, SRDiff proposes to use the diffusion process to recover just the residuals between the naively upscaled LR image and the HR, instead of directly predicting the whole image as SR3 does.

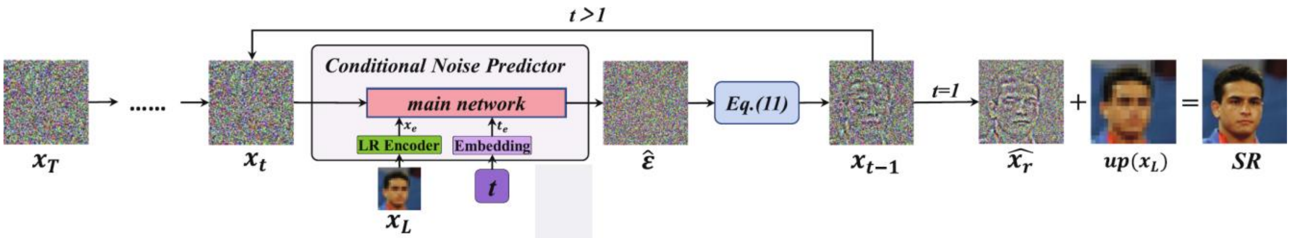


Figure 9. Schematic of the diffusion process in SRDiff. Figure extracted and edited from (H. Li et al., 2022).

The results shown in both papers seem to present sharp images with more details than other regression-based methods (i.e., the type of methods presented before this section), but authors of SR3 observe that typical metrics such as PSNR and SSIM seem to indicate that their method provides worse results. This puts into relief the problem with these pixel-based similarity scores: they do not capture the importance of structural details in real images. To prove that their results were better in terms of human acceptance, they tested the images with 50 human subjects in order to get a MOS. The tests consisted, on the one hand, in showing the LR image alongside the original HR image and the image generated by SR3, and let the user decide which one was the original HR. The second test was similar, but without showing the LR image. They then compute the *fool rate* as the fraction of trials on which a subject selects the model output over the original image.

The better training stability of DMs can also be mixed with other structures. For instance, in ResDiff (Shang et al., 2024), they combine a pre-trained SR CNN to capture the low-frequency components and a DM to capture the residuals between the ground-truth image and the predicted one. Another example is SAM-DiffSR (Wang et al., 2025), which uses the Segment Anything Model (SAM) to discern different regions on the input image and adapt the diffusion process during training to the different structures of the images, with the idea that different structures should have different noise models instead of a single global noise model as presented in DDPM.

Another relevant approach related to DMs is the Implicit Diffusion Model (IDM) (Gao et al., 2023), which introduces an implicit neural representation (INR) in the process in order to be able to attain continuous-resolution output. This basically means that, contrary to the majority of methods presented, they allow attaining arbitrary scaling factors from the same network, even non-integer factors (e.g., x1.3 scaling). An INR represents an image as a continuous function represented by a neural network that maps *coordinates* of the image to *signal* (in this case, color/intensity) values. In IDM, they represent the HR image with this INR via a multi-layer perceptron (MLP) decoder, and use a DM to condition such MLP to the LR image and the desired scale. The explicit inclusion of the scale factor in the process makes the network adjust how much it relies on LR content vs generate new content.

Despite the better perceptual quality achieved by DM and their more stable training, they still have the drawback of requiring many iterative denoising/sampling steps at inference time, resulting in slow inference. While some methods try to reduce the number of steps starting from a *degraded* image rather than pure noise, they still need many iterations. The latest approaches in SR using DM try to deal with this problem and make this number of steps as minimal as possible. For instance, in ResShift they reduce the number of diffusion steps by, instead of transitioning from HR image to pure noise, transitioning the HR image to the corresponding LR by shifting the residuals between them. Being such residual the difference between the (upsampled) LR and the HR images, the forward chain gradually shifts this residual from 0 to the full residual, moving like this from the LR to the HR image in a small number of steps rather than going all the way to Gaussian noise. Following on from this work, SinSR (Yufei Wang et al., 2024) aim to push the inference cost further to a single sampling step. The authors achieve that by applying *knowledge distillation* by training a student model that learns how to do the SR in a single step from the *teacher* model that does the whole ResShift mapping.

2.3.4 Image Transformers

Image transformers adapt the transformer architecture, originally designed for text and powering tools such as ChatGPT, to work with images. Contrarily to text, where we have a discrete sequence of words, images are a continuous 2D array of pixels. Therefore, the original transformer definition is not directly applicable to images. Instead of *words*, Vision Transformers (ViT) (Dosovitskiy et al., 2021) divide the image into small *patches*, which are effectively treated as words in a sentence. Each patch is flattened into a vector and projected into an embedding space, converting like this a 2D patch into a 1D representation that the transformer architecture can deal with. This process consists in passing each flattened patch through a (learnable) neural network that computes this embedding. In order to maintain information regarding the location of the patch in the image, positional encoding is also concatenated to each patch embedding. The sequence of patch embeddings then goes through the standard transformer layers, the most important of which is s*elf-attention*. In Self-attention each patch can *attend* (i.e., extract information) every other patch, learning relationships in the entire image. This is an important improvement over CNNs, since they have a *local* receptive field, and struggle in capturing long-term relationships between different parts of the image (they can only achieve this by doing deeper and deeper sequential convolutions). Instead, transformers can capture global, long-range, dependencies from the first layer.

The first method dealing with the super-resolution problem using transformers can be found in the Image Processing Transformer (IPT) (Chen et al., 2021). As the name of the method suggests, in this paper the super-resolution is just one of the applications of the IPT. In fact, the IPT model features multiple heads and tails tailored to different tasks, built around a shared transformer body that includes both an encoder and a decoder. Input images are first transformed into visual features, then divided into patches that act as visual tokens for further processing. High-quality images are finally reconstructed by combining the output patches.

While IPT proves useful, it is computationally demanding. The self-attention mechanism of a ViT like the one in IPT presents a large complexity that scales quadratically with image size. Moreover, the ViT architecture uses global self-attention, which lacks the locality and hierarchy that CNNs provided, by focusing on low-level features first to then gradually capturing high-level features. In SwinIR (Liang et al., 2021), they adapt the Swin Transfomer (ST) model presented in (Liu et al., 2021) to mainly overcome such limitations. Instead of computing self-attention globally across the entire image, ST divides the image into small windows and performs self-attention within each window. This reduces the complexity of the attention part, and keeps a local context, akin to what CNNs' receptive fields do. Of course, this prevents finding global interactions within the image. To enable cross-window communication, ST uses a shifted-window attention mechanism. This consists in applying self-attention within the sub-windows in which the image is divided in one layer, and shifting the windows by half their size in the next layer. This shifting makes each window overlap with its neighboring regions, making the information *flow* between them, and attaining like this global information mixing across different windows. In SwinIR (Liang et al., 2021) their network has three clearly separated parts (see Figure 10). It starts with an initial shallow feature extraction using a CNN, that captures low-level initial features from the LR image. The output of such CNN is then passed through a series of Residual Swin Transformer Blocks, each being mainly composed of a ST layer as presented before, followed by a residual connection and a small convolution layer. Finally, they combine the low-level and high-level features retrieved by the two mentioned modules and use an upsampling mechanism (e.g., sub-pixel convolution) to recover the HR image out of the feature maps.

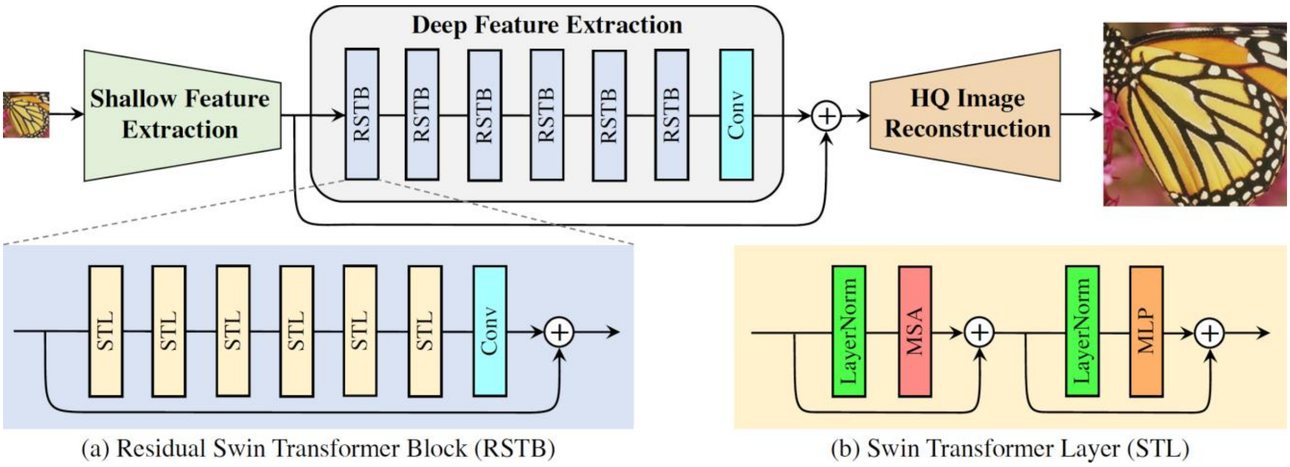


Figure 10. SwinIR architecture. Figure extracted from (Liang et al., 2021).

Since its publication, the base SwinIR architecture became very popular in the literature, and was extended in many ways. With the aim of reducing computational cost, Swin2SR (Conde et al., 2023) uses the Swin Transformer V2 architecture (Liu et al., 2022), which poses many improvements over the original ST for both training stability and performance in large models. In order to improve the global context of the original proposal, SwinFIR (Zhang et al., 2023) replaces standard convolution layers with Fast Fourier convolutions, which have an image-wide receptive field. Other methods like the Hybrid Attention Transformer (HAT) (Chen et al., 2023) try to give the model access to more useful spatial information while keeping computation manageable. To this end, it uses a hybrid attention mechanism that combines the window-based self-attention of SwinIR with channel attention, which is basically the same process used in the *Attention Networks* presented in the CNNs section above. Moreover, it introduces the Overlapping Cross-Attention (OCA) modules. In a ST, the information flow between adjacent windows is indirect (i.e., only happening because we shift the window in some layers) and slow (because of this shifting, a pixel at the border of one window does not interact *directly* with the one next to it in a neighboring window, it does it after several concatenated layers). OCA tackles this by enabling information exchange between overlapping neighboring windows. HAT defines a larger attention region that includes small overlaps from its neighboring windows. Within this larger region, *queries* come from the center window, while *keys* and *values* come from the overlapping region (neighbor + center). This introduces cross-attention, since information flows across window borders in a single layer, not only across stacked shifted windows.

2.4 Application of Deep Learning Super-Resolution to Bathymetric data

From the review in the previous section, it is clear that, for the vast majority of SR methods, their application field is optical (a.k.a. color) images. Therefore, such networks usually have three different input/output channels. However, images are not input to the model as pure integers limited to the 0..255 images' 8-bit depth range: they are usually adapted to a floating-point type via some *normalization* procedure. Therefore, as long as a compatible normalization function is applied to the input one can use the same algorithms as for optical images. Note also that, unless one wants to create a network that utilizes other sources of information, the bathymetry map is a single-channel raster, as opposed to the typical 3-channel setup of optical images. Therefore, the number of parameters for networks using solely bathymetry should be significantly lower than their image-based counterparts.

Given the complexity of gathering new samples on certain areas, having a proper way to increment the resolution of some maps in a realistic fashion is much desirable in the bathymetric domain. Unfortunately, there are few SR references applying methods other than the classic bilinear/bicubic interpolation. For instance, only a poster (Einsidleg et al., n.d.; Elmore et al., 2017) was found highlighting that using Random Forests interpolation improves results over using the classical methods, but no proper peer-reviewed article supports their claims. Despite its seemingly straightforward applicability, the use of Deep Learning SR applied to bathymetric data was not extensively exploited in the literature. In fact, to the best of our knowledge, only 8 references specifically deal with such a specific problem (Cai et al., 2023; HIDAKA et al., 2021; Irisawa and Iiyama, 2024; Ito, 2019; X. Li et al., 2022; Murakami et al., 2025; Sonogashira et al., 2020; Zhang et al., 2022). Since the main objective of the EMODnet project and this review is to discern how AI can improve bathymetric data, in the following, the findings of each of these publications are reviewed in chronological order. Also a summary of the methods is provided in Table 1.

Ref	Year	Model					Training Data		
		Proposed Architecture	**Evaluated (against)**	**In LR size**	**Out HR size**	**SR Scale**	**HR Source**	**LR Source**	**Number of LR/HR pairs**
(Ito, 2019)	2019	*None*, experimental survey paper	ESRGAN with generators: SegNet, ResNet, DCSCN, SRResNet	32	256	x8	GEBCO_2019	GEBCO_2019	40,000
(Sonogashira et al., 2020)	2020	SRResNet	Bicubic interpolation	128	256	x2	GEBCO_2019	GEBCO_2014	434
(HIDAKA et al., 2021)	2021	*None*, experimental survey paper	SRCNN, FSRCNN, ESPCN, SRGAN, ESRGAN	32	64	x2	MBES data from Okinawa Trough, gridded to 50m	MBES data from Okinawa Trough, gridded to 100m	13,723
(X. Li et al., 2022)	2022	SRResNet with transfer data training.	SRResNet trained with different loss functions	128	512	x4	*BM*: CelebA images *TL*: GEBCO_2021	*BM*: CelebA images *TL*: ETOPO-1	202,599 image 2000 bathy

(Zhang et al., 2022)	2022	DEM-SRNet, derived from EDSR, trained first on land data then transfer learning with bathymetry data.	SRCNN, VDSR, ESRGAN, EDSR	32	160	x5	*BM*: GEBCO_2021 (land areas) *TL*: GEBCO_2021 (bathymetry areas)	*BM*: NASADEM. *TL*: regional ocean DEMs (various sources).	579,900 land 28,500 bathy
(Cai et al., 2023)	2023	StFeT, seabed terrain feature extraction module and efficient transformer module.	Bicubic, SRCNN, SRGAN, SRResNet	128	128	x4	GEBCO_2022	GEBCO_2022, subsampled 4x	6,386
(Murakami et al., 2025)	2025	ESPCN trained with different data augmentation procedures.	The different data augmentation procedures	16	64	x4	MBES data from Okinawa Trough, gridded to 12,5m	MBES data from Okinawa Trough, gridded to 50m	11,053 original 154,742 DA

Table 1. Summary of the SR methods applied to bathymetry, listing the proposed architectures, against what methods were compared in the original references, the in/out sizes of the models, and the training data used. Since some of the references are just experimental survey papers, where there is no new proposal but just a comparison between different methods trained on new data, we list all the methods compared in the "Evaluated (against)" column. In the training data section, for the models using a transfer learning approach we label as BM (Base Model) the original model trained with non-bathymetric data, and with TL (Transfer Learning) the model trained from bathymetric data having the BM as starting point. Also, the DA in the last entry stands for Data Augmentation.

The first article dealing with Deep Learning SR applied to bathymetry is from 2019 (Ito, 2019). It is a brief article, not giving many precise details, and written in Japanese, therefore limiting its audience. Nonetheless, the authors trained a GAN architecture, where they played with different generator networks such as SegNet, ResNet, DCSCN (Yamanaka et al., 2017) and SRResNet, coupled with the discriminator network of ESRGAN. The dataset used was part of GEBCO_2019 (GEBCO Bathymetric Compilation Group 2019, 2019), and was composed of around 40,000 training pairs of 32x32 low resolution and 256x256 HR images. This implies that the networks were trained to apply a x8 scale factor.

In (Sonogashira et al., 2020) they trained a SRResNet, which is the generator part of the ESRGAN method. The authors claim that the full GAN proposed in ESRGAN sometimes produces fake structural artifacts and increases errors with respect to ground truth data. However, they do not provide results on the article supporting such a claim. They train on GEBCO_2014 and GEBCO_2019 (GEBCO Bathymetric Compilation Group 2019, 2019) data around the Japanese coast, pairing 434 images of 128x128 LR and 256x256 HR (i.e., a x2 scale factor). To overcome the lack of data, they propose to perform data augmentation consisting of flipping and rotations multiples of 90 degrees. Regarding the normalization procedure, they normalize the input to be between 0 and 1 based on the minimum and maximum values of each sample. As opposed to the normalization procedures used when working with optical images, the normalization proposed changes per sample, so the authors must record such min/max values for each raster in order to revert the SR results back to metric depth values. Regarding their experiments, they compare against bicubic interpolation in terms of PSNR and SSIM.

Then in 2021 an applied survey is found (HIDAKA et al., 2021) where the SRCNN, FSRCNN, ESPCN, SRGAN and ESRGAN methods are tested with bathymetric data. These methods were trained on pairs created by cropping 64×64 HR grids (50 m) with corresponding 32×32 LR grids (100 m), attaining a x2 scaling factor. With the help of data augmentation (not specified) they produced 13,723 image pairs, splitting 80% for training and 20% for testing. The methods were compared according to MSE, PSNR and DSSIM, and the performance of each method was also tested against different terrain types (gentle slope, rugged areas and flat terrain). Their conclusions indicate that the GAN-based models sometimes introduce artificial noise, and that ESRGAN consistently produced perceptually-superior results. They also claim that, while worse in some cases, the performance of SRCNN, ESPCN, and ESRGAN were mostly stable across different terrain types.

From 2022, Li et al. (X. Li et al., 2022) also resort to using just the generator part of ESRGAN (SRResNet) as in Sonogashira et al. (Sonogashira et al., 2020). However, they propose many improvements related to training and validation of such network. First, instead of training the network from scratch as proposed in this later reference, they rely on applying transfer learning. Transfer learning consists in using an already trained network for another domain than what it was originally trained for with the idea that, instead of starting training from scratch, you can transfer the knowledge the model has already learned from one domain to speed up or improve learning in another domain. In their case, they train the network on the well-known CelebFaces Attributes (CelebFacesA) dataset, consisting of faces of celebrities, and use the learned weights as starting point when training with a set of 2000 training pairs consisting of GEBCO for HR (512x512) and ETOPO-1 for LR (128x128). The main drawback of using networks already trained for color images is that, since they have 3 channels, in order to apply them to bathymetric data the single-channel depths raster must be expanded to 3 channels by duplication. Regarding normalization, they normalize in the [0, 1] range using the global max/min across all sample images. Another relevant contribution is the use of different loss functions that are combined during training. They propose to use the *content loss* function, which is the one presented originally for SRResNet, and consists of a combination of the typical MSE loss and a loss computed from differences in VGG layers (which should capture structural details). On the other hand, they also propose a *coastal* and a *water loss*. On the one hand, the *water loss* is used to penalize the generation of positive values in ocean areas, where all elevation values should be negative. On the other hand, the *coastal loss* gives more relevance to data on coastal zones, as the authors suggest that such areas have better high-resolution details due to the fact that it is far easier to obtain denser soundings than on deep waters. Due to the use of this new *water loss*, in addition to evaluating their results according to MSE and PSNR, they also introduce a *wrong pixel ratio* (WPR) metric, counting the number of cells whose values are wrongly signed with respect to their land/water label. Another interesting evaluation they perform which is not found in any other reference is checking the values of only non-interpolated GEBCO cells against their inferred HR counterparts. This is relevant because, although used for training, many cells in GEBCO are interpolated, and do not necessarily reflect real soundings of the area.

Also in 2022, Zhang et al. (Zhang et al., 2022) propose a new architecture they name DEM-SRNet derived from EDSR (Lim et al., 2017). The main change is that the number of residual blocks was expanded from the 32 in EDSR to 42. They also rely on transfer learning, but in this case the initial model was trained with land terrain datasets, which are far more numerous and available, and closer to the bathymetry domain in nature. Therefore, they collected land data from GEBCO_2021 dataset as LR (32x32) and same areas from NASADEM (OpenTopography, 2021) land samples for HR (160x160, downsampled to 3 arc-second from original at 1 arc-second), and trained an initial version of the model with 579,900 land pairs. Also, they test the model on TanDEM-X (Vassilaki and Stamos, 2020) data to check their accuracy. While they do not specify the specific sources for bathymetry, they use 28,500 samples of this type to refine the initial land-trained model. By applying the resulting model to the whole GEBCO 2021 dataset, they generate the GDEM_2022 dataset as a public product (Zhang et al., n.d.).

In 2023, Cai et al. (Cai et al., 2023) presented the most advanced network structure we found in the literature dealing with bathymetry SR. In their work, they propose the Seabed Terrain Feature Extractor Transformer (STFET) model. STFET uses a ResNet model with two interesting additions. First, they include a transformer model, which allows long-term dependencies to be established between similar regions of the terrain, improving the reconstruction. Second, they change the traditional convolutional layers by *deformable* convolution layers (Dai et al., 2017). The authors expose that the traditional convolutional layers of CNNs do not consider the influence of distance pixels adequately. This deformable convolution operation has the ability of flexibly modifying the sample positions of the convolution kernel to align with irregular features, improving like this the extraction of local seabed features. Regarding training data, they used 6386 HR-LR pairs created from GEBCO_2022 (GEBCO Bathymetric Compilation Group 2020, 2020) tiles of size 128x128 as HR input, and generated the LR versions by reducing the size by 4 and applying bicubic interpolation to get also 128x128 tiles. Note that this is the first model that assumes the input as being already upscaled to the output size, akin to pre-upsampling methods commented in Section 2.3. The method was compared against SRCNN, SRGAN and SRResNet in terms of RMSE, MAE and maximum error against a selected test set, with the proposed method attaining the best results in all cases.

The only method published in 2024, from Irisawa and Iiyamafrom (Irisawa and Iiyama, 2024), is not exactly a SR method as presented in this document. Instead of working with raster data, the authors propose to use a point set upsampling method on the raw point-based bathymetric soundings. Due to the method differing much from the methods dealt with in this report, no further details will be explored (it is also excluded from the Table 1 summary), but it is mentioned here for reference as an interesting different approach to super-resolution that could be explored in the future.

Finally, in 2025, up to the redaction of this document, the only method published is that of Murakami et al. (Murakami et al., 2025). In this reference, they test the ESPCN (Shi et al., 2016) model on bathymetric data, and follow a different approach to transfer learning as compared with previously presented proposals. In this article, they train the model using data of a specific seafloor area, and claim that, if the terrain characteristics are similar, such a model can be used to apply SR in a completely different area on the globe. Therefore, the authors use training data originating from the *Central Okinawa Trough* area, and apply the resulting model to the *Okinotorishima Islands*, which the authors claim to have similar terrain characteristics. The model was trained on multibeam data (private datasets), with 16x16 LR samples (50m resolution) and 64x64 HR correspondences (12.5m resolution). Interestingly, the code for their data augmentation procedure is available at Zenodo (matsuokadaisuke, 2024).

To sum up, it is clear from this initial review that the topic of DL SR has room for improvement in the bathymetric domain. Compared with the list of DL SR methods described in the previous section, there are still many methods that can be explored and adapted to the bathymetric domain in order to try to improve results compared to the state of the art. Also, one of the main problems to overcome is the lack of training data, and it can be seen in the *Training* data section of Table 1 that each author dealt with this problem in a different way, with diverse typologies of data and dataset sizes. In this sense, one can also see in Table 1 that it is difficult to assess the effectiveness of each proposal objectively, as there is no reference SR bathymetric dataset that all methods can benchmark against, and each method is training for attaining a different scaling factor depending on their needs, so they are not directly comparable against each other.

2.5 Conclusions and Future Work within EMODnet Bathymetry

In light of the findings of this review on the workings of the state-of-the-art DL SR methods applied to RGB images, it is clear all methods seem applicable to other types of rasters such as bathymetric maps. While some of the SR methods in the state of the art have been applied to bathymetric data, the number of references dealing with this specific problem is small (7 papers found in this review), and they test several of the most relevant approaches, up to the ESRGAN method, leaving room for **testing** also **some of the newer architectures** that appeared in recent years. For instance, the effectiveness of new architectures currently dominating the state of the art, such as SwinIR (Liang et al., 2021), were not tested in the bathymetry literature. Also related to the type of architectures available, none of the references working with bathymetry define models allowing for multiple output scaling factors, a property much desirable for practical use of such tools.

In this sense, it is planned by Coronis to **develop a Python package implementing** several of the state-of-the-art methods to test them within the same unified framework, as well as creating a **benchmark dataset** using EMODnet bathymetry data that can serve as a common ground for benchmarking the performance of the methods. This is important since, as shown in Table 1, state-of-the-art references are testing their methods on different datasets and with different model parameters (scale factors and in/out sizes), and this renders comparisons based on PSNR or other metrics not straightforward between articles.

Related to the prior point, another big problem of SR applied to bathymetry is the **lack of training data**. In the EMODnet Bathymetry project there are several sources of data to create the benchmark dataset mentioned above. For instance, one can take the global DTM product and apply down-scaling techniques to have several pairs of LR/HR raster, or have the high-resolution DTMs as base. However, in cases where such data is not enough, or one wants to train models for input/output sizes and scaling factors not aligned with such datasets, the problem of lacking data might also be faced. While data augmentation can be applied as in (Sonogashira et al., 2020) to mitigate this issue, there might be two ways to deal with this problem within EMODnet Bathymetry:

- Use *transfer learning*: instead of training a deep learning network from scratch, one can pass the weights learned from optical image SR models as initial weights when training on bathymetric data models. The network is already trained to get realistic images, so by imposing these initial weights one is just making it better at getting bathymetric maps specifically.
- Use *synthetic terrain data*: procedural terrain simulation is always a hot topic in the modelling and gaming industry. One can take advantage of such *fake worlds* generation methods to generate synthetic maps that can then be downscaled to generate LR/HR image pairs, so that they can be used to train the networks.

In conclusion, DL SR presents several research opportunities that are worth exploring within the EMODnet Bathymetry project that may benefit the bathymetric community as a whole.

3 Coastal Gap Filling - Inpainting

In the stage 2 processing, when building the EMODnet gridded bathymetry from the regional composites, the remaining gaps in the dataset are filled. In deeper waters, interpolation and GEBCO are used for this purpose, but in shallow coastal waters, this can lead to considerable inconsistencies. To provide a reasonable estimate of the bathymetry, DELTARES has invested in a prototype that takes inspiration from a classic problem for generative AI approaches, known as inpainting, to enhance the current practice. The goal of this approach is to use the surrounding bathymetry and a land-sea mask as contextual information to provide estimates of the bathymetry in the data void in a manner that is consistent with both. It should be noted that, like other generative AI methods, the result depends upon a random generation process. So, if applied twice, it will generate different results, both consistent with the surrounding bathymetry and land-sea mask. The primary aim is to emphasize generated bathymetry data that is realistic looking over estimates that approximate some overall statistic of the data but are less realistic, as will be expanded on below.

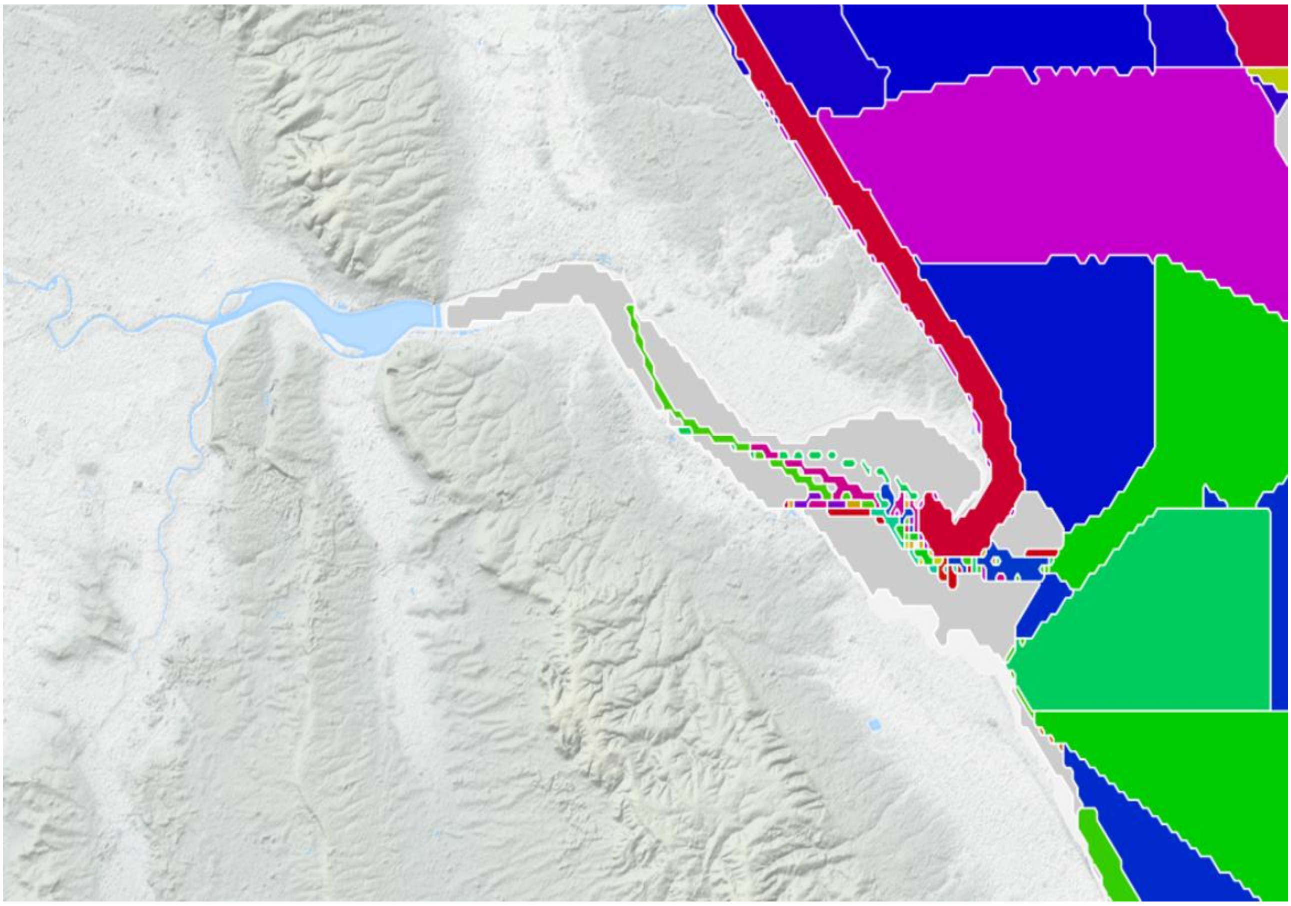


Figure 11: Example for Humber estuary showing the source IDs. For the grey areas, no surveys were available, and these were filled with values from GEBCO.

Generative models tackle this problem by treating data estimation as a problem in probability theory. A good generative model approximates the full probability distribution of all possible bathymetry realizations given the context of the surroundings. This approximated probability distribution is then sampled from to generate one or multiple instances of bathymetry data. State of the art generative AI techniques are dominated by diffusion models. These work by iteratively transforming the unknown and hard to sample from data probability distribution to some known, easy to sample from probability distribution (typically a Gaussian distribution), a process which the AI model is learned to undo. Data is generated by first sampling from the known distribution, followed by iterating over the trained model to get a sample from the data probability distribution (see Figure 12). The process of transforming the data and back has an elegant connection to (stochastic) differential equations, which has served as inspiration for efficient sampling techniques.

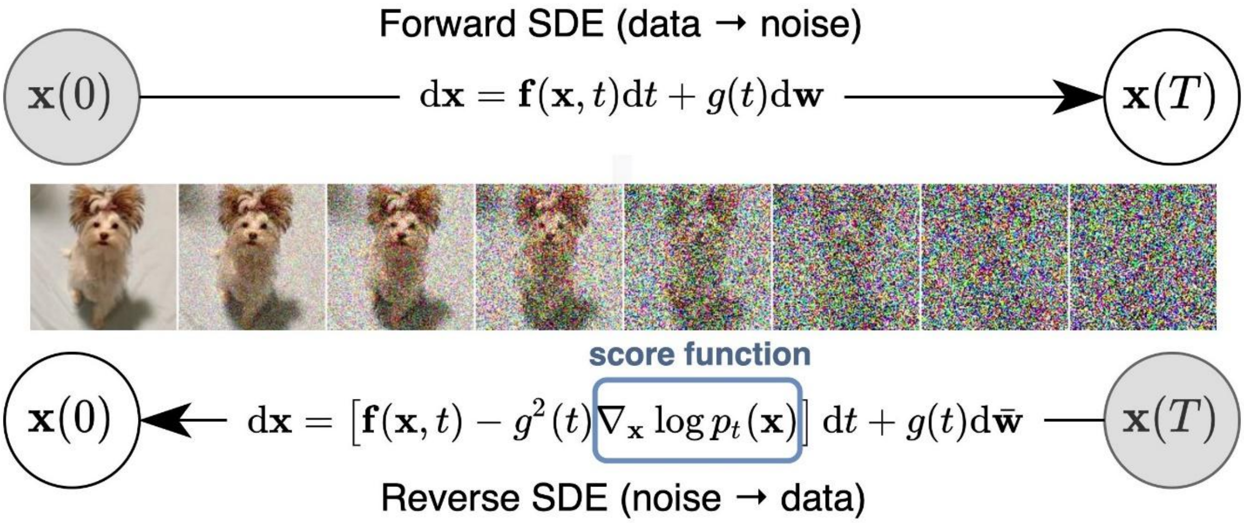


Figure 12: Schematic of the diffusion process, where a sample x(0) from the data probability distribution p_0 is transformed to a sample x(T) of the known distribution p_T. The transformation process is characterized by forward and reverse stochastic differential equations, which are solved to generate data from distribution p_0. Source: Song et al. (2021)

Additionally, the particular problem of inpainting, is an example of an inverse problem. In such a problem, part of the data is missing or masked and the goal is to reconstruct or estimate the missing data. The main feature of this problem is that the masking removes information, meaning that there are many solutions which all correspond to the same masked ground truth data. To break the tie between all these possible solutions, metrics are often employed which fall into one of two categories: distortion metrics and perceptual metrics. Generally speaking, the former measures how close the reconstructed data is to the ground truth data, while the latter tries to assess whether the reconstructed is valid (i.e. realistic looking) without comparison to some ground truth. These two classes of metrics have been shown to be in a trade-off with each other when solving an inverse problem (Blau and Michaeli, 2018; Delbracio and Milanfar, 2023) , i.e. it is impossible to both minimize distortion and maximize perception. Minimizing distortion (e.g. minimizing L2, or MSE loss to the ground truth) will often suffer from a regression-to-the-mean problem, where the final solution is a weighted sum of all possible solutions, which may not even be valid member of the dataset (it looks unrealistic). Likewise, maximizing perception by sampling from the posterior data probability distribution p(X | Y=y), i.e. the distribution of possible solutions x given observed y, can generate solutions that, while realistic looking, can be quite far from the ground truth. In this task EMODnet Bathymetry will opt for generating realistic-looking data by sampling the posterior distribution. To do this the prior distribution p(x) will be parametrized using a score-based diffusion model (Sohl-Dickstein et al., 2015; Song et al., 2021; Ho et al., 2020), and sample from the trained diffusion model to fill in the mask in the masked data y.

Diffusion models have seen widespread use in data generation (Yang et al., 2023) and solving inverse problems (Daras et al., 2024; Chung et al., 2025), most prominently in the computer vision domain (Li et al., 2023; Chen et al., 2025). In that context, inverse problems include deblurring, super-resolution, and denoising, though the discussion will be limited here to inpainting as it is most relevant to this task. Other fields where diffusion models have been applied include medical imaging (Kazerouni et al., 2023; Hein et al., 2025), earth-observation (Sanguigni et al., 2023; Liu et al., 2024; Xiao et al., 2025), time-series analysis (Yang et al., 2024; Meijer and Chen, 2024), weather and climate science (Lam et al., 2023; Chen et al., 2023; Li et al., 2024; Mardani et al., 2025), fluid dynamics (Shu et al., 2023; Wan et al., 2023; Du et al., 2024), and seismology (Wei et al. 2024; Deng et al, 2024). In the discussion below some results will be highlighted that are considered relevant to the particular inpainting problem from various domains.

Image inpainting has a long history in computer vision with most generative approaches having been applied to it (for an overview see e.g. (Xiang et al., 2023; Yang and Ruhaiyem, 2024)), and diffusion models are no exception. One of the additional benefits diffusion models have is that they can be trained unconditionally, meaning without knowledge during training of the inpainting task (in contrast to conditional training where inpainting masks are provided during training). As such, when it comes to diffusion models, inpainting applications fall into two categories: conditional and unconditional models.

One of the conditional models/approaches to highlight is Palette (Saharia et al., 2022a; Saharia et al., 2022b). Here the model is conditioned during training by simply appending the masked or degraded image to the training target. Notably, no embedding layers are used for this conditioning, as is commonly the case with other types of conditioning. Latent diffusion (Rombach et al., 2022) takes the approach to first feed the input images through an autoencoder, reducing the dimensionality of the problem, before the diffusion steps. This additional compression step is particularly useful when using large images, for both conditional and unconditional strategies. Ambient diffusion (Daras et al., 2023) offers a way to successfully train a model when only degraded data is available. This is achieved by degrading the inputs even further with a similar type of corruption and training the model on the additionally corrupted data. This of course does require prior knowledge of the corruption present in the data, automatically making it a conditional strategy.

Unconditional approaches to inpainting were already employed by Song et al. (2021) who opted to mix in a noisy version of the known parts of the image in the sampled image during every step of the sampling process. This will often lead to coherence issues at the boundaries between the known and originally masked parts of the sampled image. Various solutions have been developed to mitigate these coherence issues. Notably these include approaches where the diffusion process is moved forward again with the newly mixed image (Lugmayr et al., 2022), more sophisticated estimates of how the known and unknown parts of the image should be mixed (Wang et al., 2022), and a Bayesian approach to jointly optimize for the unknown and known parts during sampling (Zhang et al., 2023). All these approaches have in common that training the model is fairly straightforward (since it is unconditional) but sampling is increasingly complex with more advanced strategies. In general it should be noted that unconditional models are more flexible since they can be applied to multiple inverse problems, but typically perform worse than conditional models.

Diffusion models have been applied to inverse problems on a wide variety of data structures. In the field of medical imaging, it is often the case that the input data and the to be reconstructed image do not live in the same data space. For instance, the input and target data can differ in dimensionality (2D vs 3D data), or the spaces are related through some transformation. The diffusion model is then also responsible for the translation between data spaces. Examples include DVG-Diffusion, which reconstructs 3D CT scans from 2D X-ray slices (Xie et al., 2025), and SURE-Score, which reconstructs multi-coil MRI images where inputs and targets are related through Fourier transformations (Aali et al., 2023). Diffusion models have already seen a wide variety of applications in medical imaging and the reader is referred to more comprehensive overviews (Kazerouni et al., 2023; Hein et al., 2025).

Another area where diffusion models have seen recent successes is remote sensing/satellite imagery (Liu et al., 2024), particularly the problem of images obscured by clouds typically interpreted as an inpainting or reconstruction task. Zou et al. (2024) take a conditional approach, using a condition embedding for the cloudy images to guide the training and sampling processes (Zou et al., 2024). However, since satellite imagery often features fine details, reconstructions are particularly vulnerable to the distortion vs perception trade-off. To combat this, various approaches have been proposed. The DDPM-CR approach performs an additional step of feature extraction on the output of the diffusion model which is then fed to the actual model that does the cloud removal (Jing et al., 2023). Zhao et al. (2023) propose a method that employs both temporal image sequences and additional modes of data (SAR in this case) to enhance the quality of the final image (Zhao and Jia, 2023). Barth et al. (2024) use the probabilistic nature of diffusion models to provide ensembles for the related task of reconstructing satellite-derived chlorophyll-a concentrations (Barth et al., 2024). They also use a strategy reminiscent of Ambient Diffusion to counter the lack of cloud-free images.

More recently, in a problem very similar to the reconstruction of bathymetry data, diffusion models have been applied to filling voids in digital elevation models. The authors of Diff-DEM train a conditional model where they generate a new inpainting mask at each training step and append it to the network input, resulting in a model that outperforms interpolation methods used more traditionally for DEMs (Lo and Peters, 2024). Panangian and Bittner (2025) improve on this with Dfilled by using optical satellite imagery as guidance, generating more realistic masks using Perlin noise, and using an additional refinement network (Panangian and Bittner, 2025).

These examples highlight another aspect in improving the performance of diffusion models: incorporating contextual data other than the inputs and targets. Embedding contextual data has arguably become synonymous with diffusion models, given the success and fame of text-to-image applications where textual information is used to guide image generation (Nichol et al., 2022; Esser et al., 2024). However, this is not the only application where contextual information can be efficiently used. In particular, data reconstruction from sparse observations is an area worth highlighting, where the focus will be on meteorology and fluid dynamics.

Incorporating sparse observations in climate data can be interpreted as an inpainting task as done by various papers (Chao et al., 2025; Tsao et al., 2025). The diffusion models are trained unconditionally on the climate data, and during sampling the observational data is first spatially smoothed before merging with the network output to ensure the point observations are properly incorporated into the 2D outputs. The same approach can also be taken for the weather simulations themselves, i.e. data assimilation. The authors of DiffDA similarly apply an inpainting strategy plus spatial smoothing to assimilate point observational data into a diffusion model for weather forecasting, GraphCast (Lam et al., 2023; Huang et al., 2024a). A different approach to incorporating sparse data is by embedding the observations as contextual information. To train a diffusion model to solve fluid dynamics partial differential equations, Zhuang et al. (2025) explore various methods of embedding sparse observations by doing a Voronoi based interpolation of the data (Zhuang et al, 2025). Du et al. (2024) take this approach a step further by doing the diffusion steps in a latent space encoded by conditional neural fields (Du et al., 2024). Additionally in the realm of solving differential equations, diffusion models can be made physics informed by guiding the model during sampling with the gradient of the residual. Shu et al. (2023) combine this with aforementioned inpainting strategies for sparse observations to learn fluid dynamics (Shu et al., 2023). DiffusionPDE improves on this by treating the observational data in a similar way as the residual to guide the sampling process (Huang et al., 2024b).

As a final remark, it should be noted that the above is not an exhaustive summary of diffusion models applied to inpainting/data reconstruction nor an exhaustive list of domains where they have been employed to that end. The aim for this review is to give an idea of the current state and the dynamics around this problem. Newly sophisticated training and sampling methods typically emerge from the field of computer vision, which are then taken up and adapted to other scientific disciplines. This was the case for diffusion models themselves and the most common strategies for image inpainting. Adaptations to particular scientific disciplines usually then comes in the form of making optimal use of structure in the relevant data or processing pipelines, as in the case of medical imaging, clever choice of the data the condition the model with, as in the case of satellite imagery and DEM reconstruction, or guiding the model inference with additional rules the output should obey, as in the case of solving differential equations.

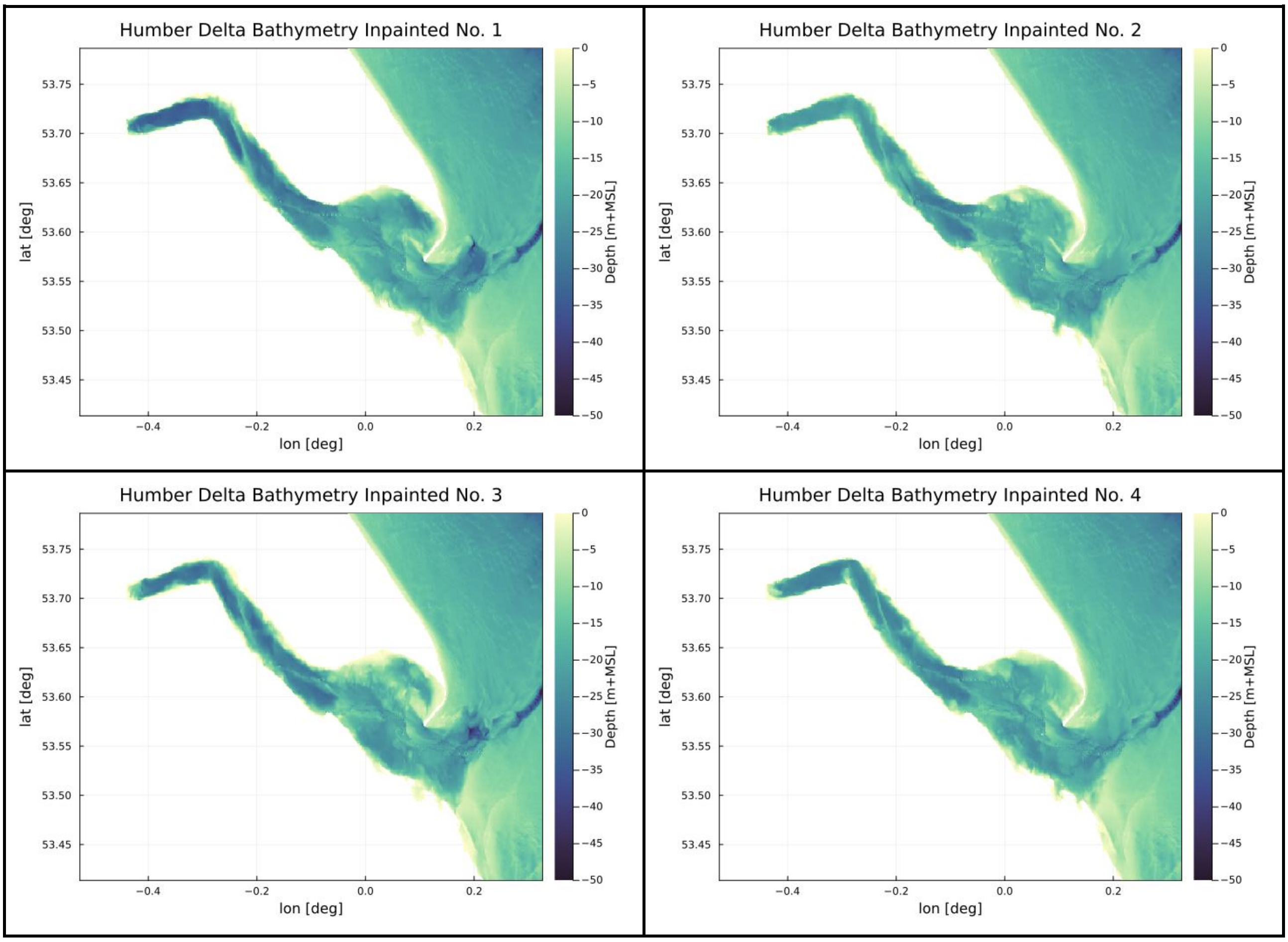


Figure 13: Four AI generated bathymetries for the Humber estuary, generated using the previously developed prototype, where all GEBCO filled points were replaced with generated values.

3.1 Findings and recommendations

Diffusion models have established themselves as the state-of-the-art in image generation due to their high quality output and the ability to be guided by additional data types. Inspired by this success, diffusion models have recently seen application in a wide range of scientific domains for data generation, particularly for inpainting related tasks. To this end, the base concepts of a diffusion model have been adapted to the particular case at hand, ranging from architectural changes to the diffusion model itself to improved strategies for incorporating contextual information. Applications to bathymetric data appear to be still lacking. Recent applications to similarly structured data, such as digital elevation models (Lo and Peters, 2024; Panangian and Bittner, 2025), indicate that diffusion models for bathymetric data might prove fruitful. To that end, EMODnet Bathymetry plans to further develop and improve a prototype model for bathymetry inpainting developed by Deltares in the previous EMODnet phase (see Figure 13).

In part based on previous experience, two main challenges are foreseen in applying diffusion models to bathymetry. The first is model validation. Choosing proper metrics to evaluate model performance is not obvious due to the aforementioned trade-off between perceptual and distortion metrics (Blau and Michaeli, 2018; Delbracio and Milanfar, 2023). While not fully solving this trade-off, the probabilistic nature of diffusion models opens up additional evaluation metrics via ensemble reconstructions (Barth et al., 2024), by e.g. evaluating biases in the generated ensemble.

The second challenge foreseen is the (limited) availability of training data. There are several approaches to tackle the limited availability, most prominently data augmentation and transfer learning. Additionally, sufficient contextual data might restrain the model enough to be properly trained on smaller datasets. The contextual data can come in various forms, such as non-bathymetric data about the region of interest, or non-gridded bathymetric measurements in the area. Applications in various scientific domains provide examples and inspiration on how to incorporate such heterogeneous data sources.

In conclusion, it is believed that diffusion models for data generation and inpainting are worthwhile exploring further in the context of bathymetry data, and can offer added benefit to the EMODnet bathymetry project.

4 Error detection

4.1 Overview

Currently "errors" can occur in the global DTM which arise out of multiple reasons from datum or coordinate shifts to biased input datasets (see Figure 14). The data processing pipeline for obtaining the global DTM product of EMODnet Bathymetry involves many actors, ranging from the individual data providers, who process the raw data, passing by the regional coordinators, who compose larger maps combining the individual ones, and ending in the global coordinator, creating the Europe-wide map merging the regional areas. Therefore EOMAP works on a tool that automatically detects different types of artifacts to warn the user and prevent their propagation. AI pattern classification solutions were and are currently evaluated to deal with common defects in the map, such as outliers, gridding artifacts, patterns caused by reprojection errors, etc.

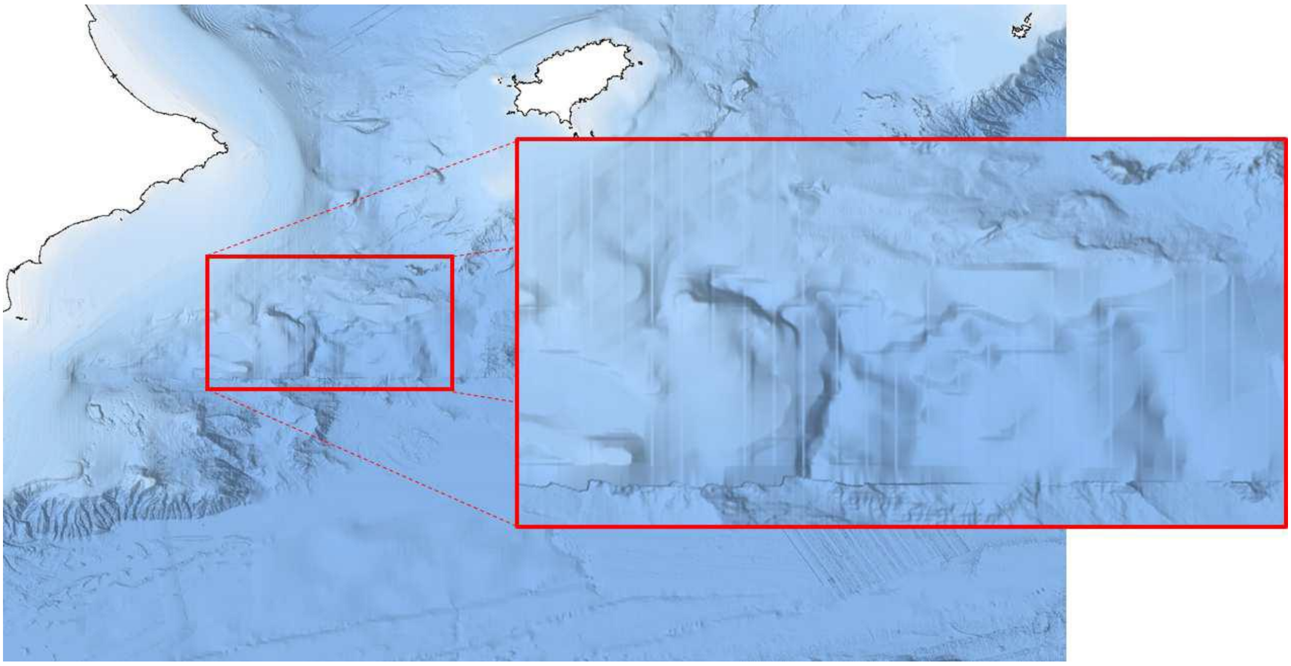


Figure 14: An example of a possible artifact, resulting from adding a DTM which underwent a wrong conversion between reference systems

The overall objective is the development of an AI/ML solution to highlight and identify potential bias/artefacts in bathymetric surfaces (DTMs) to increase quality of data and reduce manual action. To reach this goal the following way forward was elaborated:

1) EMODnet basin coordinators provide the top 20+ surfaces where bias has been spotted in the past
2) Based on 1) a catalogue of bias/artefacts will be created and categorized according to the detected errors. The bias will be labeled and a database for it will be built.
3) Performance of a technical evaluation on different methods and potential outcomes
4) Re-evaluate this with the basin coordinators
5) Based on 2-4) EOMAP will train the ML/AI methods

Next to the data input provided by the basin coordinators the collaborative virtual environment, which is fed with artefacts, will be used as a data source. Furthermore, artificial errors might be generated.

4.2 Technical approach

The proposed workflow starts with a systematic categorization of artefacts in the bathymetric grids. The goal is to decide whether it can be reliably detected by deterministic/statistical filters or if it requires a learned non-linear AI-method. For this purpose, it is advantageous to look at existing work on DEM. These already offer solutions to various similar phenomena, and the data is quite close to bathymetric grids. Various kinds of artefacts could be identified and categorized in Hirt et al. (2018) as seen in Figure 15. The shown DEM anomalies include phenomena like spike, line (horizontal and vertical) as well as patch defects.

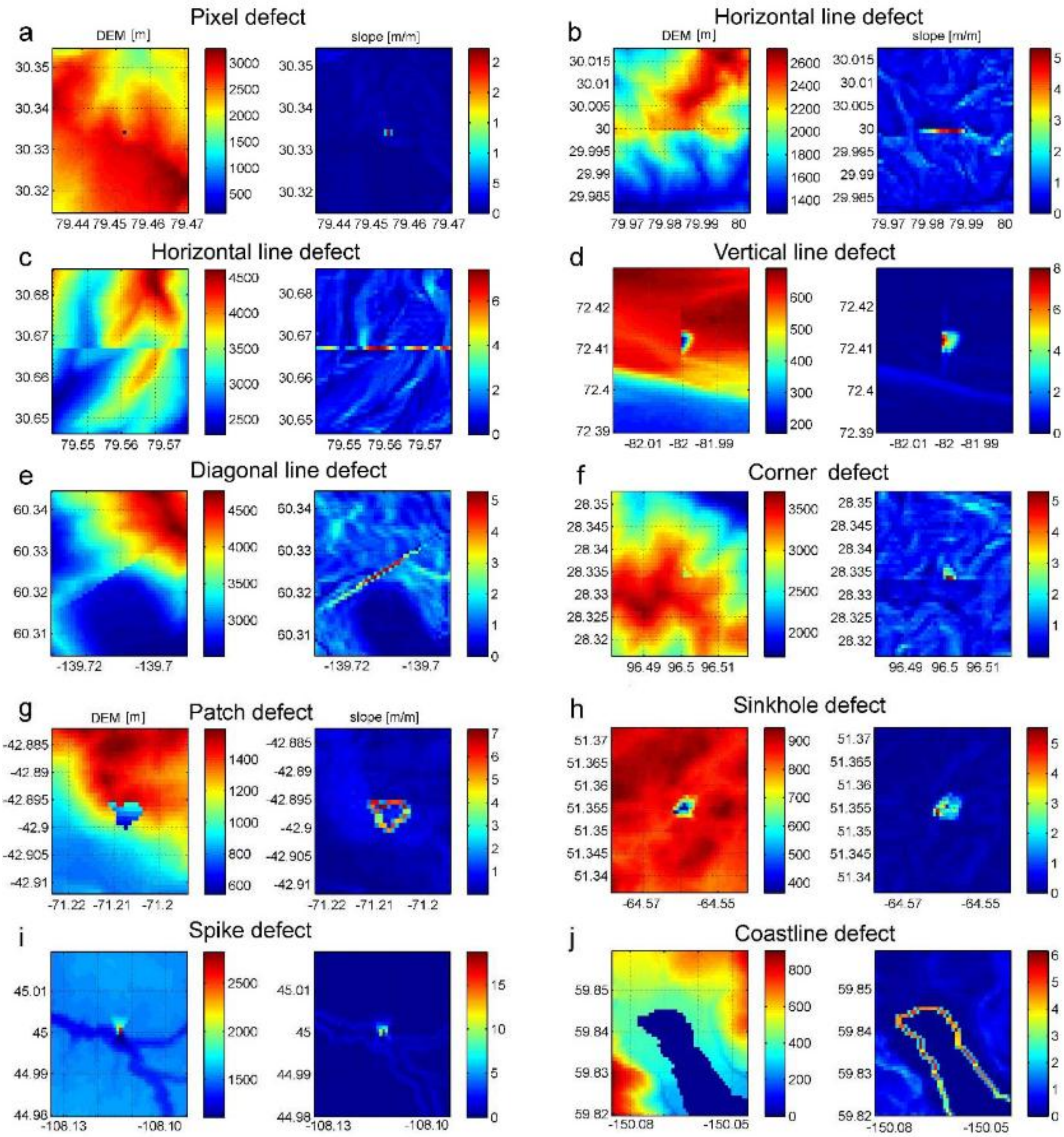


Figure 15: Examples for various kinds of anomalies in DEMs from Hirt et al. 2018

In practice, artifacts that are spatially directional or clearly linked to the acquisition geometry are good candidates for statistical methods, whereas complex spatially heterogeneous anomalies (e.g. subtle multibeam artefacts) are better suited to AI-based detections.

Statistical Methods often employ terrain-derivatives like slope, curvature or roughness to spot defects. A promising method for the mentioned defects is the Maximum Slope Approach (MSA) from Hirt et al. (2018). It uses the sensitivity of terrain slope for sudden steps in the DEM, which are also applicable for bathymetric grids and supports the identification of sharp non-geomorphic breaks such as interpolation seams or unnaturally straight boundaries. A different approach would be to employ spectral filtering to address striping artifacts done in Arrell et al. (2008). The method uses the spectral domain of an image to filter out spikes which correspond to the striping regularity in the image. The advantage here is that, in the best-case scenario, the filter only touches the stripes and, unlike other low-pass filters, prevents any details from being lost.

For artifacts that require a learning-based approach, an AI object detection setup is adopted. Given that the labelled bathymetric artefacts are scarce the training strategy should combine (i) heavy data augmentation (geometric transforms, intensity/relief perturbations etc.), which has been shown to significantly improve performance (Shorten et al., 2019), (ii) transfer learning with fine-tuning of pre-trained detectors on sonar or other relevant data and (iii) synthetic training data in which clean bathymetric surfaces are overlaid with anomalies to increase the training samples (Schlüter et al., 2022).

It is suggested in the absence of domain specific pretrained models to use state of the art object detection networks like *DINO-DETR* (Zhang et al., 2022) or *YOLOv11* (Khanam et al., 2024), which can be pretrained on vast datasets like Object365 from Shao et al. (2019). This combination offers a valid strategy to detect and localize non-linear anomalies present in the bathymetric grid.

4.3 Findings and recommendations

The first challenge is to identify, group and catalogue the artefacts to understand its nature and severity. Categorize each type to be dealt with either statistical methods or a ML-Detector is needed. The foreseeable second challenge is the limited availability of training data. To tackle that data augmentation methods should be applied and plant synthetic artefacts into the data for training. For the final step one would resort to highly popular Object-Detector Networks like *DINO-DETR* which should be capable, given enough data, to localize the anomalies.

It is also recommended to keep close contact with the basin coordinators through the whole process.

5 Conclusions and planned follow-up

In summary, three steps are identified at various stages of the bathymetry processing chain where it is believed that AI applications can provide additional benefit, based on surveys of the bathymetric domain and related scientific domains. Based on these surveys plans have been outlined to develop a pilot for each of the three applications, and identified some initial challenges that need to be overcome.

For the super-resolution use case, there is a pre-existing body of bathymetric applications to draw from, in addition to work in closely related subjects. Current bathymetric applications however do not make use of the state-of-the-art super-resolution techniques found in other fields. Therefore, it is proposed to develop a unified framework in which several deep learning super-resolution techniques can be applied and compared on the same bathymetry data. This common ground enables the proper benchmarking of the various deep learning approaches, something which has so far been difficult due to performance being measured on different datasets.

A body of work applying the state-of-the-art in gap filling (or inpainting) on bathymetry data seems not to exist currently. However a survey of related scientific domains reveals a wide swath of applications and outlines several strategies which could be used in the bathymetry domain. An initial pilot was already initiated during the previous EMODnet phase, and it is proposed to further build on this to properly investigate the added benefit of deep learning for gap filling of bathymetry data. This particular technique can benefit from additional, non-bathymetry data on the region of interest, for which experts will be consulted on the relevant data sources.

Similarly, in the case of error detection one can draw inspiration from related fields. In this particular application, not all detection schemes require deep learning approaches. Existing statistical methods are fully capable of detecting and correcting some artifact types. Only the more complex artifacts will benefit from an AI based detection scheme. For these types of artifacts, using pre-trained object detection models offer a good starting point. To establish a database of artifacts to use as a starting point, consultation with data providers or regional coordinators is suggested.

The application to bathymetry data is the common thread to these otherwise distinct AI applications, and as such the availability of sufficient training data is a common challenge to the three proposed pilots. The mitigation strategies outlined for the three pilots share techniques, such as data augmentation and the use of synthetic data. It is therefore within the realm of possibilities that one solution to the availability of training data might benefit all three applications. Follow-up work will need to determine whether this is actually the case. Finally, it is emphasized that the quality of the training data is crucial for training AI models. Luckily, EMODnet bathymetry has an excellent provenance tracking and system of quality labels, even at the pixel level.

6 References

Aali, A., Arvinte, M., Kumar, S., Tamir, J.I., 2023. Solving inverse problems with score-based generative priors learned from noisy data, 57th Asilomar Conference on Signals, Systems, and Computers, pp. 837-843

Ahn, N., Kang, B., Sohn, K.-A., 2018. Fast, Accurate, and Lightweight Super-Resolution with Cascading Residual Network, in: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Eds.), Computer Vision – ECCV 2018, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 256–272. https://doi.org/10.1007/978-3-030-01249-6_16

Al-Mekhlafi, H., Liu, S., 2023. Single image super-resolution: a comprehensive review and recent insight. Front. Comput. Sci. 18, 181702. https://doi.org/10.1007/s11704-023-2588-9

Akerele, T. M., Owoigbe, K. V., & Idowu, R. L. The Role of Artificial Intelligence in Automating Bathymetric Data Analysis and Feature Detection. 2025 https://doi.org/10.5281/zenodo.15869266

Arrell, K., Wise, S., Wood, J., & Donoghue, D. 2008.

Spectral filtering as a method of visualising and removing striped artefacts in digital elevation data. Earth Surface Processes and Landforms: The Journal of the British Geomorphological Research Group, 33(6), 943-961.

Barth, A., Brajard, J., Alvera-Azcárate, A., Mohamed, B., Troupin, C., Beckers, J.M., 2024. Ensemble reconstruction of missing satellite data using a denoising diffusion model: application to chlorophyll-a concentration in the Black Sea, Ocean Science 20(6), pp. 1567-1584

Blau, Y., Michaeli, T., 2018. The Perception-Distortion Tradeoff, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6228-6273. http://doi.org/10.1109/CVPR.2018.00652

Burt, P., Adelson, E., 1983. The Laplacian Pyramid as a Compact Image Code. IEEE Trans. Commun. 31, 532–540. https://doi.org/10.1109/TCOM.1983.1095851

Cai, W., Liu, Y., Chen, Y., Dong, Z., Yuan, H., Li, N., 2023. A Seabed Terrain Feature Extraction Transformer for the Super-Resolution of the Digital Bathymetric Model. Remote Sens. 15, 4906. https://doi.org/10.3390/rs15204906

Chao, J., Pan, B., Chen, Q., Yang, S., Wang, J., Nai, C., Zheng, Y., Li, X., Yuan, H., Chen, X., Lu, B., 2025. Learning to infer weather states using partial observations, Journal of Geophysical Research: Machine Learning and Computation, 2(1), pp. 2024JH000260

Chen, H., Wang, Y., Guo, T., Xu, Chang, Deng, Y., Liu, Z., Ma, S., Xu, Chunjing, Xu, Chao, Gao, W., 2021. Pre-Trained Image Processing Transformer, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 12294–12305. https://doi.org/10.1109/CVPR46437.2021.01212

Chen, H., Xiang, Q., Hu, J., Ye, M., Yu, C., Cheng, H., Zhang, L., 2025. Comprehensive exploration of diffusion models in image generation: a survey, Artificial Intelligence Review, 58(4), pp. 99.

Chen, S., Long, G., Jiang, J., Liu, D., Zhang, C., 2023. Foundation models for weather and climate data understanding: a comprehensive review, arXiv preprint arXiv:2312.03014

Chen, X., Wang, X., Zhou, J., Qiao, Y., Dong, C., 2023. Activating More Pixels in Image Super-Resolution Transformer, in: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 22367–22377. https://doi.org/10.1109/CVPR52729.2023.02142

Choi, J.-S., Kim, M., 2017. A Deep Convolutional Neural Network with Selection Units for Super-Resolution, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1150–1156. https://doi.org/10.1109/CVPRW.2017.153

Chung, H., Kim, J., Ye, J. C., 2025. Diffusion Models for Inverse Problems, arXiv preprint arXiv:2508.01975.

Conde, M.V., Choi, U.-J., Burchi, M., Timofte, R., 2023. Swin2SR: SwinV2 Transformer for Compressed Image Super-Resolution and Restoration, in: Karlinsky, L., Michaeli, T., Nishino, K. (Eds.), Computer Vision – ECCV 2022 Workshops. Springer Nature Switzerland, Cham, pp. 669–687. https://doi.org/10.1007/978-3-031-25063-7_42

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y., 2017. Deformable Convolutional Networks, in: 2017 IEEE International Conference on Computer Vision (ICCV). Presented at the 2017 IEEE International Conference on Computer Vision (ICCV), pp. 764–773. https://doi.org/10.1109/ICCV.2017.89

Daras, G., Shah, K., Dagan, Y., Gollakota, A., Dimakis, A., Klivans, A., 2023. Ambient Diffusion: Learning clean distributions from corrupted data, Advances in Neural Information Processing Systems, 36, pp. 288-313

Dars, G., Chung, H., Lai, C.-H., Mitsufuji, Y., Ye, J. C., Milanfar, P., Dimakis, A. G., Delbracio, M., 2024. A Survey on Diffusion Models for Inverse Problems, arXiv preprint arXiv:2410.00083.

Delbracio, M., Milanfar, P., 2023. Inversion by direct iteration: an alternative to denoising diffusion for image restoration. Transactions on Machine Learning Research

Deng, F., Wang, S., Wang, X., Fang, P., 2024. Seismic data reconstruction based on conditional constraint diffusion model, IEEE Geoscience and Remote Sensing Letters, 21, pp. 1-5

Dong, C., Loy, C.C., He, K., Tang, X., 2016a. Image Super-Resolution Using Deep Convolutional Networks. IEEE Trans. Pattern Anal. Mach. Intell. 38, 295–307. https://doi.org/10.1109/TPAMI.2015.2439281

Dong, C., Loy, C.C., Tang, X., 2016b. Accelerating the Super-Resolution Convolutional Neural Network, in: Leibe, B., Matas, J., Sebe, N., Welling, M. (Eds.), Computer Vision – ECCV 2016. Springer International Publishing, Cham, pp. 391–407. https://doi.org/10.1007/978-3-319-46475-6_25

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. https://doi.org/10.48550/arXiv.2010.11929

Du, P., Parikh, M.H., Fan, X., Liu, X.Y., Wang, J.X., 2024. Conditional neural field latent diffusion model for generating spatiotemporal turbulence, Nature Communications 15(1), pp. 10416

Einsidleg, D., Nock, K., Bonanno, D., Elmore, P., Wood, W., Petry, F., Smith, L., n.d. High Resolution Bathymetry Estimation Improvement with Single Image Super-Resolution Algorithm “Super-Resolution Forests.”

Elmore, P.A., Nock, K., Bonanno, D., Smith, L., Ferrini, V.L., Petry, F.E., 2017. Enhancing Deep-Water Low-Resolution Gridded Bathymetry Using Single Image Super-Resolution. Presented at the AGU Fall Meeting Abstracts, pp. OS31C-1420.

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., 2024. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis, International Conference on Machine Learning, PMLR 235, pp. 12606-12633

Fan, Y., Shi, H., Yu, J., Liu, D., Han, W., Yu, H., Wang, Z., Wang, X., Huang, T.S., 2017. Balanced Two-Stage Residual Networks for Image Super-Resolution, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Honolulu, HI, USA, pp. 1157–1164. https://doi.org/10.1109/CVPRW.2017.154

Gao, S., Liu, X., Zeng, B., Xu, S., Li, Y., Luo, X., Liu, J., Zhen, X., Zhang, B., 2023. Implicit Diffusion Models for Continuous Super-Resolution, in: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10021–10030. https://doi.org/10.1109/CVPR52729.2023.00966

GEBCO Bathymetric Compilation Group 2019, 2019. The GEBCO_2019 Grid - a continuous terrain model of the global oceans and land. https://doi.org/10.5285/836F016A-33BE-6DDC-E053-6C86ABC0788E

GEBCO Bathymetric Compilation Group 2020, 2020. The GEBCO_2020 Grid - a continuous terrain model of the global oceans and land. https://doi.org/10.5285/A29C5465-B138-234D-E053-6C86ABC040B9

Grabbert, P., Bothe, M., & Westfeld, P. (2024). Spatiotemporal analysis of satellite imagery using AI-based optical hydrography for adaptive planning of hydrographic resources. *The International Hydrographic Review*, *30*(2).

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep Residual Learning for Image Recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778. https://doi.org/10.1109/CVPR.2016.90

Hein, D., Bozorgpour, A., Merhof, D., Wang, G., Physics-inspired generative models in medical imaging, 2025. Annual Review of Biomedical Engineering, 27(1), pp. 499-525.

HIDAKA, M., MATSUOKA, D., KUWATANI, T., KANEKO, J., KASAYA, T., KIDO, Y., ISHIKAWA, Y., KIKAWA, E., 2021. Super-resolution for Ocean Bathymetric Maps Using Deep Learning Approaches : A Comparison and Validation. Geoinformatics 32, 3–13. https://doi.org/10.6010/geoinformatics.32.1_3

Hirt, C. 2018. Artefact detection in global digital elevation models (DEMs): The Maximum Slope Approach and its application for complete screening of the SRTM v4. 1 and MERIT DEMs. *Remote Sensing of Environment*, *207*, 27-41

Ho, J., Jain, A., Abbeel, P., 2020. Denoising diffusion probabilistic models, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20. Curran Associates Inc., Red Hook, NY, USA, pp. 6840–6851.

Horn, R.A., Johnson, C.R., 1985. Matrix Analysis. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511810817

Hu, Y., Gao, X., Li, J., Huang, Y., Wang, H., 2018. Single Image Super-Resolution via Cascaded Multi-Scale Cross Network. https://doi.org/10.48550/arXiv.1802.08808

Huang, D., Liu, H., 2015. A Short Survey of Image Super Resolution Algorithms. J. Comput. Sci. Technol. Updat. 2, 19–29.

Huang, L., Gianinazzi, L., Yu, Y., Dueben, P.D., Hoefler, T., 2024a. DiffDA: A Diffusion Model for weather-scale Data Assimilation, International Conference on Machine Learning, PMLR 235, pp. 19798-19815

Huang, J., Yang, G., Wang, Z., Park, JJ., 2024b. DiffusionPDE: Generative PDE-solving under partial observations, Advances in Neural Information Processing Systems, 37, pp. 130291-130323

Irisawa, N., Iiyama, M., 2024. High-Resolution Bathymetry by Deep-Learning Based Point Cloud Upsampling. IEEE Access 12, 4387–4398. https://doi.org/10.1109/ACCESS.2023.3349149

Ito, K., 2019. Efficient Bathymetry by Learning-Based Image Super Resolution. Fish. Eng. 56, 47–50. https://doi.org/10.18903/fisheng.56.1_47

Jakobsson, M., Mayer, L., Coakley, B., Dowdeswell, J. A., Forbes, S., Fridman, B., ... & Weatherall, P. (2012).

The international bathymetric chart of the Arctic Ocean (IBCAO) version 3.0. *Geophysical*

Research Letters, *39*(12).

Jing, R., Duan, F., Lu, F., Zhang, M., Zhao, W., 2023. Denoising Diffusion Probabilistic feature-based network for cloud removal in Sentinel-2 imagery, Remote Sensing, 15(9), pp. 2217

Kazerouni, A., Aghdam, E. K., Heidari, M., Azad, R., Fayyaz, M., Hachihaliloglu, I., Merhof, D., 2023. Diffusion models in medical imaging: A comprehensive survey, Medical Image Analysis, 88, pp. 102846

Khanam, R., & Hussain, M. (2024).

Yolov11: An overview of the key architectural enhancements. *arXiv preprint arXiv:2410.17725*.

Kim, J., Lee, J.K., Lee, K.M., 2016a. Accurate Image Super-Resolution Using Very Deep Convolutional Networks, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1646–1654. https://doi.org/10.1109/CVPR.2016.182

Kim, J., Lee, J.K., Lee, K.M., 2016b. Deeply-Recursive Convolutional Network for Image Super-Resolution, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Las Vegas, NV, USA, pp. 1637–1645. https://doi.org/10.1109/CVPR.2016.181

Lai, W.-S., Huang, J.-B., Ahuja, N., Yang, M.-H., 2017. Deep Laplacian Pyramid Networks for Fast and Accurate Super-Resolution, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Honolulu, HI, pp. 5835–5843. https://doi.org/10.1109/CVPR.2017.618

Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Alet, F., Ravuri, S., Ewalds, T., Eaton-Rosen, Z., Hum W., Merose, A., Hoyer, S., Holland, G., Vinyals, O., Stoot, J., Pritzel, A., Mohamed, S., Battaglia, P., 2023. Learning skillful medium-range global weather forecasting, Science 382(6677), pp. 1416-1421.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., Shi, W., 2017. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 105–114. https://doi.org/10.1109/CVPR.2017.19

Li, H., Yang, Y., Chang, M., Chen, S., Feng, H., Xu, Z., Li, Q., Chen, Y., 2022. SRDiff: Single image super-resolution with diffusion probabilistic models. Neurocomputing 479, 47–59. https://doi.org/10.1016/j.neucom.2022.01.029

Li, L., Carver, R., Lopez-Gomez, I., Sha, F., Anderson, J., 2024. Generative emulation of weather forecast ensembles with diffusion models, Science Advances, 10(13) pp. Eadk4489

Li, X., Li, J., Williams, Z., Huang, X., Carroll, M., Wang, J., 2022. Enhanced Deep Learning Super-Resolution for Bathymetry Data, in: 2022 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT). Presented at the 2022 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT), pp. 49–57. https://doi.org/10.1109/BDCAT56447.2022.00014

Li, X., Ren, Y., Jin, X., Lan, C., Wang, X., Zeng, W., Wang, X., Chen, Z., 2023. Diffusion Models for Image Restoration and Enhancement: A Comprehensive Survey, arxiv preprint arxiv:2308.09388.

Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R., 2021. SwinIR: Image Restoration Using Swin Transformer, in: 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). Presented at the 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), IEEE, Montreal, BC, Canada, pp. 1833–1844. https://doi.org/10.1109/ICCVW54120.2021.00210

Lim, B., Son, S., Kim, H., Nah, S., Lee, K.M., 2017. Enhanced Deep Residual Networks for Single Image Super-Resolution, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1132–1140. https://doi.org/10.1109/CVPRW.2017.151

Liu, Y., Yue, J., Xia, S., Ghamisi, P., Xie, W., Fang, L., 2024. Diffusion models meet remote sensing: principles, methods, and perspectives, IEEE Transactions on Geoscience and Remote Sensing, 62, pp. 1-22

Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., Ning, J., Cao, Y., Zhang, Z., Dong, L., Wei, F., Guo, B., 2022. Swin Transformer V2: Scaling Up Capacity and Resolution, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11999–12009. https://doi.org/10.1109/CVPR52688.2022.01170

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. Presented at the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE Computer Society, pp. 9992–10002. https://doi.org/10.1109/ICCV48922.2021.00986

Lo, K.S.H., Peters, J., 2024. Diff-dem: A diffusion probabilistic approach to digital elevation model void filling, IEEE Geoscience and Remote Sensing Letters, 21, pp. 1-5

Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R., Van Gool, L., 2022. Repaint: Inpainting using denoising diffusion probabilistic models, Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11461-11471

Mardani, M., Brenowitz, N., Cohen, Y., Pathak, J., Chen, C.-Y., Liu, C.-C., Vahdat, A., Nabian, M. A., Ge, T., Subramaniam, A., Kashinath, K., Kautz, J., Pritchard, M., 2025. Communications Earth & Environment, 6:124.

matsuokadaisuke, 2024. matsuokadaisuke/adaptive-sr: Adaptive-sr v1.0.0. https://doi.org/10.5281/ZENODO.14043132

Meijer, C., Chen, L., 2024. The rise of diffusion models in time-series forecasting, arXiv preprint arxiv:2401.03006

Murakami, K., Matsuoka, D., Takatsuki, N., Hidaka, M., Kaneko, J., Kido, Y., Kikawa, E., 2025. Adaptive Super-Resolution for Ocean Bathymetric Maps Using a Deep Neural Network and Data Augmentation. Earth Space Sci. 12, e2024EA003610. https://doi.org/10.1029/2024EA003610

Nichol, A.Q., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., Mcgrew, B., Sutskever, I., Chen, M., 2022. GLIDE: Toward Photorealistic Image Generation and Editing with Text-Guided Diffusion Models, International Conference on Machine Learning, PMLR 162, pp. 16784-16804

Odena, A., Dumoulin, V., Olah, C., 2016. Deconvolution and Checkerboard Artifacts. Distill 1, e3. https://doi.org/10.23915/distill.00003

OpenTopography, 2021. NASADEM Global 1 arc-second Digital Elevation Model. https://doi.org/10.5069/G93T9FD9

Panangian, D., Bittner, K., 2025. Dfilled: Repurposing Edge-Enhancing DIffusion for Guided DSM Void Filling, 2025 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW), pp. 526-534

Pascanu, R., Mikolov, T., Bengio, Y., 2013. On the difficulty of training recurrent neural networks, in: Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28, ICML'13. JMLR.org, Atlanta, GA, USA, p. III-1310-III–1318.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B., 2022. High-resolution image synthesis with latent diffusion models, Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10684-10695

Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D.J., Norouzi, M., 2023. Image Super-Resolution via Iterative Refinement. 2022a. IEEE Trans. Pattern Anal. Mach. Intell. 45, 4713–4726. https://doi.org/10.1109/TPAMI.2022.3204461

Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D., Norouzi, M., 2022b. Palette: Image-to-Image Diffusion Models, ACM SIGGRAPH 2022 conference proceedings, 15, pp. 1-10

Sajjadi, M.S.M., Schölkopf, B., Hirsch, M., 2017. EnhanceNet: Single Image Super-Resolution Through Automated Texture Synthesis. Presented at the 2017 IEEE International Conference on Computer Vision (ICCV), IEEE Computer Society, pp. 4501–4510. https://doi.org/10.1109/ICCV.2017.481

Sanguigni , F., Czerkawski, M., Papa, L., Amerini, I., Saux, B. L., 2023. Diffusion models for earth observation use-cases: from cloud removal to urban change detection, arXiv preprint arxiv:2311.06222

Schlüter, H. M., Tan, J., Hou, B., & Kainz, B. 2022
Natural synthetic anomalies for self-supervised anomaly detection and localization. In *European Conference on Computer Vision* (pp. 474-489). Cham: Springer Nature Switzerland.

Schultz, R.R., Stevenson, R.L., 1996. Extraction of high-resolution frames from video sequences. IEEE Trans. Image Process. 5, 996–1011. https://doi.org/10.1109/83.503915

Schultz, R.R., Stevenson, R.L., 1995. Improved definition video frame enhancement, in: 1995 International Conference on Acoustics, Speech, and Signal Processing. Presented at the 1995 International Conference on Acoustics, Speech, and Signal Processing, pp. 2169–2172 vol.4. https://doi.org/10.1109/ICASSP.1995.479905

Shang, S., Shan, Z., Liu, G., Wang, L., Wang, X., Zhang, Z., Zhang, J., 2024. ResDiff: combining CNN and diffusion model for image super-resolution, in: Proceedings of the Thirty-Eighth AAAI Conference on

Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence, AAAI'24/IAAI'24/EAAI'24. AAAI Press, pp. 8975–8983. https://doi.org/10.1609/aaai.v38i8.28746

Shao, S., Li, Z., Zhang, T., Peng, C., Yu, G., Zhang, X., ... & Sun, J. 2019.

Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 8430-8439)

Shaw, S., & Arnold, J. (1993, October). Automated error detection in multibeam bathymetry data.

In *Proceedings of OCEANS'93* (pp. II-89). IEEE.

Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z., 2016. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1874–1883. https://doi.org/10.1109/CVPR.2016.207

Shorten, C., & Khoshgoftaar, T. M. 2019.

A survey on image data augmentation for deep learning. *Journal of big data*, *6*(1), 1-48.

Shu, D., Li, Z., Fairmani, A.B., 2023. A physics-informed diffusion model for high-fidelity flow field reconstruction, Journal of Computational Physics, 478, pp. 111972

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S., 2015. Deep unsupervised learning using nonequilibrium thermodynamics, in: International Conference on Machine Learning, pp. 2256-2265.

Song, Y., Sohl-Dickstein, J., Kingma, D., Kumar, A., Ermon S., & Poole, B., 2021. Score-Based Generative Modeling through Stochastic Differential Equations, in: International Conference on Learning Representations

Sonogashira, M., Shonai, M., Iiyama, M., 2020. High-resolution bathymetry by deep-learning-based image superresolution. PLOS ONE 15, e0235487. https://doi.org/10.1371/journal.pone.0235487

Stark, H., Oskoui, P., 1989. High-resolution image recovery from image-plane arrays, using convex projections. JOSA Vol 6 Issue 11 Pp 1715-1726. https://doi.org/10.1364/JOSAA.6.001715

Tai, Y., Yang, J., Liu, X., 2017. Image Super-Resolution via Deep Recursive Residual Network, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Honolulu, HI, pp. 2790–2798. https://doi.org/10.1109/CVPR.2017.298

Tsao, V., Chaney, N.W., Veveakis, M., 2025. Probabilistic Spatial Interpolation of SparseData using Diffusion Models, arXiv preprint arXiv:2506.00033

van Ouwerkerk, J.D., 2006. Image super-resolution survey. Image Vis. Comput. 24, 1039–1052. https://doi.org/10.1016/j.imavis.2006.02.026

Vassilaki, D.I., Stamos, A.A., 2020. TanDEM-X DEM: Comparative performance review employing LIDAR data and DSMs. ISPRS J. Photogramm. Remote Sens. 160, 33–50. https://doi.org/10.1016/j.isprsjprs.2019.11.015

Wan, Z.Y., Baptista, R., Boral, A., Chen, Y.F., Anderson, J., Sha, F., Zepeda-Nuñez, L., 2023. Debias coarsely, sample conditionally: Statistical downscaling through optimal transport and probabilistic diffusion models, Advanced in Neural Information Processing Systems, 36, pp.47749-47763

Wang, C., Hao, Z., Tang, Y., Guo, J., Yang, Y., Han, K., Wang, Y., 2025. SAM-DiffSR: Structure-Modulated Diffusion Model for Image Super-Resolution. https://doi.org/10.48550/arXiv.2402.17133

Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., Loy, C.C., 2019. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks, in: Leal-Taixé, L., Roth, S. (Eds.), Computer Vision – ECCV 2018 Workshops, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 63–79. https://doi.org/10.1007/978-3-030-11021-5_5

Wang, Yufei, Yang, W., Chen, X., Wang, Yaohui, Guo, L., Chau, L.-P., Liu, Z., Qiao, Y., Kot, A.C., Wen, B., 2024. SinSR: Diffusion-Based Image Super-Resolution in a Single Step, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 25796–25805. https://doi.org/10.1109/CVPR52733.2024.02437

Wang, Y., Yu, J., Zhang, J., 2022. Zero-shot image restoration using denoising diffusion null-space model, arXiv preprint arXiv:2212.00490

Wei, X., Zhang, C., Wang, H., Tan, C., Xiong, D., Jiang, B., 2024. Seismic Data Interpolation via Denoising Diffusion Implicit Models with Coherence-Corrected Resampling, IEEE Transactions on Geoscience and Remote Sensing, 62, pp. 1-17

Xiao, A., Xuan, W., Wang, J., Huang, J., Tao, D., Lu, S., Yokoya, N., 2025. Foundation models for remote sensing and earth observation: a survey, IEEE Geoscience and Remote Sensing Magazine

Xiang, H., Zou, Q., Nawaz, M.A., Huang, X., Zhang, F., Yu, H., 2023. Deep learning for image inpainting: a survey, Pattern Recognition, 134, pp. 109046

Xie X., Liu, J., Fan, H., Han, Z., Tang, Y., Qu, L., 2025. DVG-Diffusion: Dual-view Guided Diffusion Model for CT Reconstruction from X-rays, arXiv preprint arXiv:2503.17804

Yamanaka, J., Kuwashima, S., Kurita, T., 2017. Fast and Accurate Image Super Resolution by Deep CNN with Skip Connection and Network in Network, in: Liu, D., Xie, S., Li, Y., Zhao, D., El-Alfy, E.-S.M. (Eds.), Neural Information Processing. Springer International Publishing, Cham, pp. 217–225. https://doi.org/10.1007/978-3-319-70096-0_23

Yang, J., Ruhaiyem, N.I.R., 2024. Review of Deep Learning-based Image Inpainting Techniques, IEEE Access, 12, pp. 138441-138482

Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Zhang, W., Cui, B., Yang, M.-H., 2023. Diffusion Models: a Comprehensive Survey of Methods and Applications, ACM Computing Surveys, 56(4), pp. 1-39.

Yang, Y., Jin, M., Wen, H., Zhang, C., Liang, Y., Ma, L., Wang, Y., Liu, C., Yang, B., Xu, Z., Bian, J., 2024. A survey on diffusion models for time series and spatio-temporal data, arXiv preprint arxiv:2404.18886

Zhang, B., Xiong, W., Ma, M., Wang, M., Wang, D., Huang, Xing, Yu, L., Zhang, Q., Lu, H., Hong, D., Yu, F., Wang, Z., Wang, J., Li, X., Gong, P., Huang, Xiaomeng, 2022. Super-resolution reconstruction of a 3 arc-second global DEM dataset. Sci. Bull. 67, 2526–2530. https://doi.org/10.1016/j.scib.2022.11.021

Zhang, B., Xiong, W., Ma, M., Wang, M., Wang, D., Huang, Xing, Yu, L., Zhang, Q., Lu, H., Hong, D., Yu, F., Wang, Z., Wang, J., Li, X., Gong, P., Huang, Xiaomeng, n.d. GDEM_2022.

Zhang, D., Huang, F., Liu, S., Wang, X., Jin, Z., 2023. SwinFIR: Revisiting the SwinIR with Fast Fourier Convolution and Improved Training for Image Super-Resolution. https://doi.org/10.48550/arXiv.2208.11247

Zhang, Y., Li, Kunpeng, Li, Kai, Wang, L., Zhong, B., Fu, Y., 2018. Image Super-Resolution Using Very Deep Residual Channel Attention Networks, in: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Eds.), Computer Vision – ECCV 2018, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 294–310. https://doi.org/10.1007/978-3-030-01234-2_18

Zhang, G., Ji, J., Zhang, Y., Yu, M., Jaakkola, T., Chang, S., 2023. Towards Coherent Image Inpainting Using Denoising Diffusion Implicit Models, arXiv preprint arXiv:2304.03322

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., ... & Shum, H. Y. 2022.

Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*.

Zhao, X., Jia, K., 2023. Cloud removal in remote sensing using sequential-based diffusion models, Remote Sensing, 15(11), pp. 2861

Zhuang, Y., Cheng, S., Duraisami, K., 2025. Spatially-aware diffusion models with cross-attention for global field reconstruction with sparse observations, Computer Methods in Applied Mechanics and Engineering, 435, pp. 117623

Zong, Z., Zha, L., Jiang, J., Liu, X., 2022. Asymmetric Information Distillation Network for Lightweight Super Resolution, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Presented at the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, New Orleans, LA, USA, pp. 1248–1257. https://doi.org/10.1109/CVPRW56347.2022.00131

Zou, X., Li, K., Xing, J., Zhang, Y., Wang, S., Jin, L., Tao, P., 2024. Diffcr: A fast conditional diffusion framework for cloud removal from optical satellite images, IEEE Transactions on Geoscience and Remote Sensing, 62, pp. 1-14