applied sciences

MDPI

*Article*

# Vision Transformers for Anomaly Detection and Localisation in Leather Surface Defect Classification Based on Low-Resolution Images and a Small Dataset

**Antony Douglas Smith, Shengzhi Du * and Anish Kurien**

Department of Electrical Engineering, Faculty of Engineering and the Built Environment, Pretoria 0184, South Africa; smithad@tut.ac.za (A.D.S.); kurienam@tut.ac.za (A.K.)
* Correspondence: dus@tut.ac.za

**Abstract:** Genuine leather manufacturing is a multibillion-dollar industry that processes animal hides from varying types of animals such as sheep, alligator, goat, ostrich, crocodile, and cow. Due to the industry's immense scale, there may be numerous unavoidable causes of damages, leading to surface defects that occur during both the manufacturing process and the bovine's own lifespan. Owing to the heterogenous and manifold nature of leather surface characteristics, great difficulties can arise during the visual inspection of raw materials by human inspectors. To mitigate the industry's challenges in the quality control process, this paper proposes the application of a modern vision transformer (ViT) architecture for the purposes of low-resolution image-based anomaly detection for defect localisation as a means of leather surface defect classification. Utilising the low-resolution defective and non-defective images found in the opensource Leather Defect detection and Classification dataset and higher-resolution MVTec AD anomaly benchmarking dataset, three configurations of the vision transformer and three deep learning (DL) knowledge transfer methods are compared in terms of performance metrics as well as in leather defect classification and anomaly localisation. Experiments show the proposed ViT method outperforms the light-weight state-of-the-art methods in the field in the aspect of classification accuracy. Besides the classification, the low computation load and low requirements for image resolution and size of training samples are also advantages of the proposed method.

**Keywords:** deep learning; computer vision; defect detection; leather quality; vision transformer networks; semi-supervised learning



**Citation:** Smith, A.D.; Du, S.; Kurien, A. Vision Transformers for Anomaly Detection and Localisation in Leather Surface Defect Classification Based on Low-Resolution Images and a Small Dataset. *Appl. Sci.* **2023**, *13*, 8716. https://doi.org/10.3390/app13158716

Academic Editors: Bruno Fernandes and Rogério Pontes

Received: 4 July 2023
Revised: 25 July 2023
Accepted: 25 July 2023
Published: 28 July 2023



## 1. Introduction

Leather is a widely used material in various industries such as the automotive, fashion, and furniture industries due to its unique texture, durability, and aesthetic appeal. With the global leather market valued at USD 440.64 billion and a projected growth to USD 468.49 billion in 2023 and USD 738.61 billion by 2030 [1], the necessity for optimization and efficiency in all aspects of its production is undeniable. Within the leather manufacturing industry, one of the most important processes involved in transforming raw materials into consumer goods is the inspection process or quality control. By ensuring a high level of consistency in the grading of raw and processed materials, manufacturers not only greatly reduce labour costs, but also eliminate unnecessary material losses caused by the misclassification of exported commodities. Traditional methods for leather surface defect detection rely on qualified technicians to visually inspect every inch of the processed materials. This procedure can be labour-intensive, time-consuming, and highly subjective, which increases the risk of human error and inconsistent defect identification. To overcome these challenges, there is a growing interest in leveraging artificial intelligence (AI) and computer vision techniques for automated and accurate leather surface defect detection.

In recent years, vision transformers [2] have been emerging as a powerful approach in computer vision tasks, showing superior performance in image classification [3], object detection [4], and segmentation [5]. Vision transformers are deep neural networks that process images in a patch-based manner, where the image is divided into non-overlapping patches and then linearly embedded into a sequence of vectors. These sequences of vectors are then processed by transformer layers, originally proposed for natural language processing tasks, to capture both local and global contextual information. The self-attention mechanism [6] in transformers allows for capturing long-range dependencies, making them highly effective for analysing complex patterns and structures in images.

When implementing such a system in the real-world manufacturing environment, certain practical considerations and potential challenges should be considered. To process images in real-time, a trade-off between model accuracy and speed must be considered. As manufacturers may undergo changes in equipment, surface material types, and/or processing chemicals, the ability to quickly re-train or fine-tune a model on a new dataset can be greatly advantageous. Despite the fact that low-resolution images reduce the computational demands of the model, allowing it to train faster and be applied in a real-time setting, the presence of fewer pixels representing the defective area make them difficult to accurately localise. Defects may be subtle and could potentially be missed or blurred out in a low-resolution image. As the transformer splits the image into patches, if the resolution is too low, each patch might not contain enough detail for the model to make accurate predictions. Furthermore, the manufacturing environment may include variations in lighting, viewing angle, and other factors that might change the appearance of the leather without constituting a defect. The inspection system would therefore require a controlled environment, or a high level of robustness and adaptability from the model. Finally, owing to the reduced availability and feasibility of resources such as high-definition image capturing systems and processing power, many current machine vision-based AI systems are not an option for manufacturers in developing countries, where the majority of raw leather products are produced.

For this reason, this research focusses on using anomaly detection and localisation using lower-resolution leather surface images for defect classification. Training an image classifier on a small low-resolution dataset offers several benefits. Firstly, it reduces the hardware and resource requirements for the training process. Since low-resolution images have smaller data sizes, less computational power and storage capacity are needed to process and store the data. This translates to cost savings and improved accessibility, especially for individuals or organizations with limited resources. Secondly, training on a small dataset allows for increased efficiency. With fewer images to process, the training time is significantly reduced, enabling faster iterations and experimentation with different model architectures and hyper-parameters. This accelerated development cycle facilitates the refinement of the classifier and speeds up the overall research or development process. Consequently, an image classifier on a small dataset of low-resolution images not only reduces hardware and resource requirements but also enhances efficiency, making it an advantageous approach in certain scenarios.

The contributions of this paper include the modification and implementation of two transformer-based model configurations tailored for low-resolution leather surface defect detection, as well as an in-depth analysis of the model's performance compared to existing pre-trained DL methods. This study would be beneficial for the adoption of vision transformers (ViT) in the leather industry, enabling a faster, cheaper, and more robust leather surface defect detection system for improved product quality control.

The rest of the paper is organized as follows. Section 2 provides an overview of the related work in the field. Section 3 describes the proposed method for addressing the research problem. In Section 4, we present the experiments conducted to validate the effectiveness of the proposed approach. Finally, Section 5 concludes the paper by summarizing the key findings and highlighting the contributions made in this study.

## 2. Related Work

Vision transformers [2] are deep learning models based on the transformer architecture, which have recently gained a lot of attention for their remarkable performance in various computer vision tasks. In the context of leather surface defect detection, accurate and reliable defect identification is crucial for quality control in the leather industry. Vision transformers have shown potential for achieving high accuracy and efficiency. This section reviews pertinent literature on image-based methods for surface defect detection and identification on leather and similar image domains. Also included are methods such as deep learning classification algorithms and vision transformers for the application of anomaly detection as a method of identifying and localising surface defects.

### *2.1. Surface Defect Detection Methods*

Quality control inspection is important across all manufacturing industries and has therefore had many possible solutions applied over the years. Recently, computer vision and machine learning have become the most prominent, as these methods can play a critical role in ensuring product quality, reducing wasted materials, and improving manufacturing efficiency in industries such as the automotive, aerospace, and electronics industries, where surface defects can have significant consequences. Current methodologies can be categorized into four groups, viz., statistical, structural, spectral, model-based, and machine learning.

Image processing methods can generally be classified under the following four categories: (1) statistical algorithms such as a features-based wavelet method [7], (2) structural algorithms such as local binary patterns (LBP) [8] or Gabor filters [9], (3) spectral algorithms such as the wavelet transform [10], and (4) model-based algorithms, as can be found in in the work of Wang et al. [11]. Comprehensive investigations conducted in the fields of automated visual defect detection for flat steel surfaces [12] and fabric defect detection in textile manufacturing [13] present a succinct compilation of the diverse methodologies utilized in surface defect detection. While several image processing methods are successfully utilised to detect surface anomalies for defect detection, these methods tend to be specifically tailored to the environment and surface characteristics for which the system was designed. When compared to modern machine learning techniques these methods can be less robust and adaptable when introduced to new or unexpected variations in input data.

Several influential machine learning techniques have been used to detect and identify surface defects in similar image domains such as metals, plastics, textiles, and wood. One such method is to utilise complex transfer learning methods from the pre-trained CNN models. For instance, VGG16 [14] and Xception [15] were developed for the automatic classification of powder bed defects in the selective laser sintering (SLS) process using very small datasets [16]. In the domain of surface crack detection, Tabernik et al. [17] developed a system for image acquisition, pre-processing, and segmentation-based deep learning (DL) that ultimately achieved an average precision of 99% with only 33 defective sample images. When training a DL classifier for the identification of defects in fabric, both defective and non-defective samples are required. As it can be difficult to obtain defective samples for training, Han et al. [18] proposed a semi-supervised learning method of stacked convolutional autoencoders trained on non-defective fabric samples only.

Anomaly detection methods are widely used in the field of surface defect detection to identify and classify abnormal or defective regions on the surface of various materials. These methods typically involve the use of machine learning algorithms that leverage statistical techniques, pattern recognition, or deep learning approaches to identify deviations from normal surface patterns. One common approach is based on image processing techniques that extract relevant features from surface images, followed by the use of statistical algorithms (Gaussian distribution) to detect anomalies based on deviations from the expected patterns [19]. Another approach involves the use of convolutional neural networks (CNNs) trained on large datasets of normal and defective surfaces to learn com-

plex patterns and identify anomalies in real-time [20]. Other techniques involve various methods of determining a deviation from reconstructed input images. The simplest method of achieving this is with the use of autoencoders to reduce the normal images to a discrete latent space for reconstruction. Anomalous regions will be improperly reconstructed and when compared to normal images, these regions are not only detected but also localised [21]. A slightly more complex method is to de-compose images using dual deep reconstruction networks-based image decomposition (DDR-ID) while optimizing for three losses, viz., one-class loss, latent space constraint loss, and reconstruction loss. Once trained, the DDR-ID can decompose an unseen image into residual components to determine anomaly scores. It can then, based on predefined thresholds, confidently classify anomalous images [22]. The third reconstructive method uses a multi-stage image resynthesis framework. The method takes defective images and attempts to repair suspicious regions that have large deviations from the original input image. Defects are then localised in the residual map between input images and the repaired outputs [23]. The use of an adversarial autoencoder (AAE) was suggested by Beggel et al. [24] in an effort to reduce the influence of the 'contamination' of flawed samples on autoencoder networks. A generative model of the input data can be trained by combining the reconstruction error with an adversarial training criterion, and a discriminator network gains flexibility by learning to distinguish between samples originating from the encoder.

The utilisation of DenseNet architecture in anomaly detection for surface defects detection [25] and classification [26] in fabrics has shown promising outcomes because of its capacity to capture intricate patterns and handle noisy data effectively. However, its dense architecture presents a challenge, as it demands substantial computational resources and time for both training and inference processes.

### *2.2. Leather Surface Defect Detection Methods*

Given the magnitude and longevity of the leather manufacturing industry, it is to be expected that a variety of machine vision approaches have been applied over the years to address the challenge of leather surface defect detection and classification. Many conventional image processing techniques were used, specifically in the early years (10+ years) of this application, which produced comparatively outstanding classification results. One very influential method is that of Pistori et al. [27], whereby colour and texture features are collected from a small custom leather dataset (258 images), using greyscale co-occurrence matrices (GLCM). To determine the most optimal classification method, these extracted features are used in conjunction with 10-fold cross-validation for comparison of output results between a normalized Gaussian radial basis function network, support vector machine (SVM), and k-nearest neighbours (KNN).

More recently, the success of neural networks in the field has led to several innovative solutions. Moganam and Ashok [28] used a class activation mapping technique to find the region of interest for the class of defect after popular CNNs, such as GoogLeNet, SqueezeNet, and ResNet, are trained using leather input images from six categories of defective leather sample images [29]. These same authors present the original high-resolution ($4608 \times 3288$ pixels) version of the Leather Defect detection and Classification dataset that is utilised in low-resolution format throughout this paper. Using the original dataset, the authors propose the use of a grey-level co-occurrence matrix (GLCM) to extract statistical texture features from defective and non-defective leather samples for a perceptron neural network classifier to be trained on labelled datasets of these texture features to identify five common leather defects: folding marks, grain off, growth marks, loose grain, and pinholes [30].

Advancing their previous work in this field of interest, the teams of Gan and Liong adopted a generative adversarial network (GAN) as a means of reliably synthesizing further normal images in order to augment an already limited training set and therefore improve the accuracy of feature extraction and classification of the typical AlexNet architecture [31].

Due to the ambiguous and unpredictable nature of wet-blue leather, the previous methods are all primarily focused on finished leather. However, Chen et al. [32] proposed a method for wet-blue leather using hyperspectral target detection (HTD) to suppress the background while enhancing the contrast of input images before extracting features, where three neural networks were used. A 1D-CNN focusses on spectral feature defects, while defects with spatial features are the focus of a 2D-Unet, and a 3D-Unet architecture simultaneously processes spatial and spectral information. The combination of these three DL methods allows for the extraction and capture of a wide spectrum of defective textures, shapes, and sizes.

### 2.3. Vision Transformers in Anomaly Detection

Although originally developed for the application of natural language processing (NLP) [6], the vision transformer (ViT), a type of deep learning architecture, has recently gained attention for its promising applications in image anomaly detection. The architecture is a deep learning model for image classification that replaces traditional convolutional layers with self-attention mechanisms. It divides an image into non-overlapping patches, linearly projects them into flat vectors, and uses a transformer encoder to capture global contextual information. The resulting feature vectors are then passed through multiple layers of feed-forward neural networks to predict class labels. The vision transformer architecture has achieved state-of-the-art performance on various image classification benchmarks, demonstrating the power of self-attention in capturing long-range dependencies in images without relying on convolutional operations.

Rapid advancements in the field can be attributed to a series of highly influential papers describing the evolution of modified NLP transformers and the adaptation of self-attention mechanisms [6] for implementation in image-oriented applications [2,33–35]. Several vision transformer variations have been created as a result of these publications and the numerous machine-vision-based fields of research. One of the earliest transformer models specifically adapted to anomaly detection is the vision-transformer-based image anomaly detection and localisation network (VT-ADL) [36]. The ViT is a network designed to work on image patches, trying to preserve their positional information. Using these fundamental characteristics, a modified transformer network known as the VT-ADL was developed that can learn the unique and diverse features of the normal data in a semi-supervised way (normal data only) to localise anomalous regions using Gaussian approximation.

The ViTAE model [37] is a tailored vision transformer model that uses spatial pyramid reduction modules to embed input images into tokens with multi-scale context to learn robust feature representations for objects at different scales. It also includes a convolution block in parallel with the self-attention module in each transformer layer, enabling it to learn local features and global dependencies together. This gives the model both scale invariance and locality-inductive biases, making it a powerful model for image analysis.

Further variants of the vision transformer model worth mentioning include the Swin-ViT [38] and MiM Vanilla-ViT [39].

### 2.4. Transfer Learning Methods

For comparative training metrics and classification results, transfer learning from three popular pre-trained deep learning (DL) architectures is performed on the same leather detection and classification dataset [29]. All three models are pre-trained on the ImageNet dataset [40].

The first pre-trained network architecture to be utilised is the residual neural network (ResNet) [41]. The ResNet-50 model was selected as the preferred variant due to its optimal balance between performance and reduced computational requirements. The mid-range ResNet-50 model, as depicted in the architecture diagram presented in Figure 1, was chosen over ResNet-101 and ResNet-152 due to its decreased model size and complexity,

which helps minimize memory requirements and enables faster convergence, training, and inference.

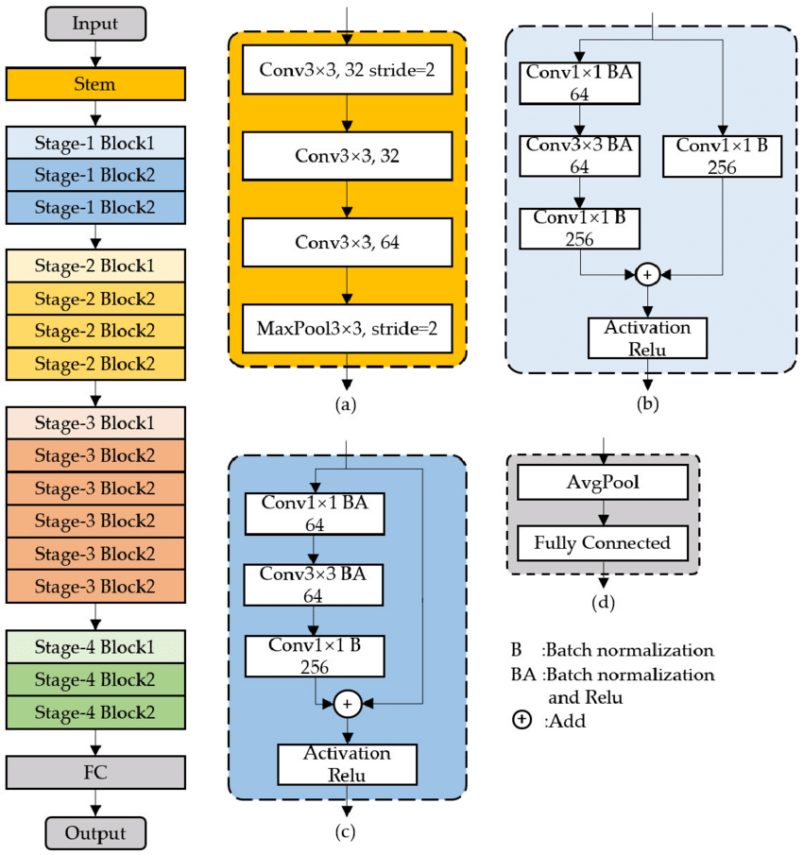



**Figure 1.** ResNet-50 architectural block diagram. (**a**) Stem block; (**b**) Block1; (**c**) Block2; (**d**) FC-Block.

Figure 1 presents a detailed representation of the ResNet-50 network [42]. The network displayed on the left side of Figure 1 comprises the stem module, four residual modules, and a fully connected neural network layer. The numerical annotations, 32, 64, and 256, in the figure signify the corresponding number of convolution channels. The notation "B" indicates the batch normalization operation, while "BA" denotes the combination of batch normalization and relu activation. Stage-2 through Stage-4 are the same as Stage-1, except for an increased number of iterations of block2. Lastly, the model's final FC-block utilizes an average pooling layer and a fully connected layer.

The second DL model, depicted in Figure 2, is the Inception-V3 architecture [43], which shares similar elements with the ResNet-50 model despite being slightly more complex. Succinctly stated by Google Cloud [44], the structure comprises convolutions, average pooling, max pooling, concatenations, dropouts, and fully connected layers. Batch normalization is used extensively throughout the model and is applied to activation inputs, while loss is computed using the Softmax activation function.

The third architecture applied, EfficientNet-B0, is described as the smallest and least complex variant of the EfficientNet series [45]. Though this lightweight version of the model may not be as accurate as the high-end EfficientNet-B7 version, it was designed with the idea of achieving a balance between model accuracy and computational efficiency by scaling the depth, width, and resolution of the network. As such, it has the advantage of lower computational requirements and therefore significantly faster training and inference times, which are essential to real-world applications. Another risk in using deeper and more complex architectures is the increased risk of overfitting, especially when working with smaller datasets. A simplified block diagram of the EfficientNet-B0 model is depicted in Figure 3; however, the 237-layer model itself is far more complex internally.

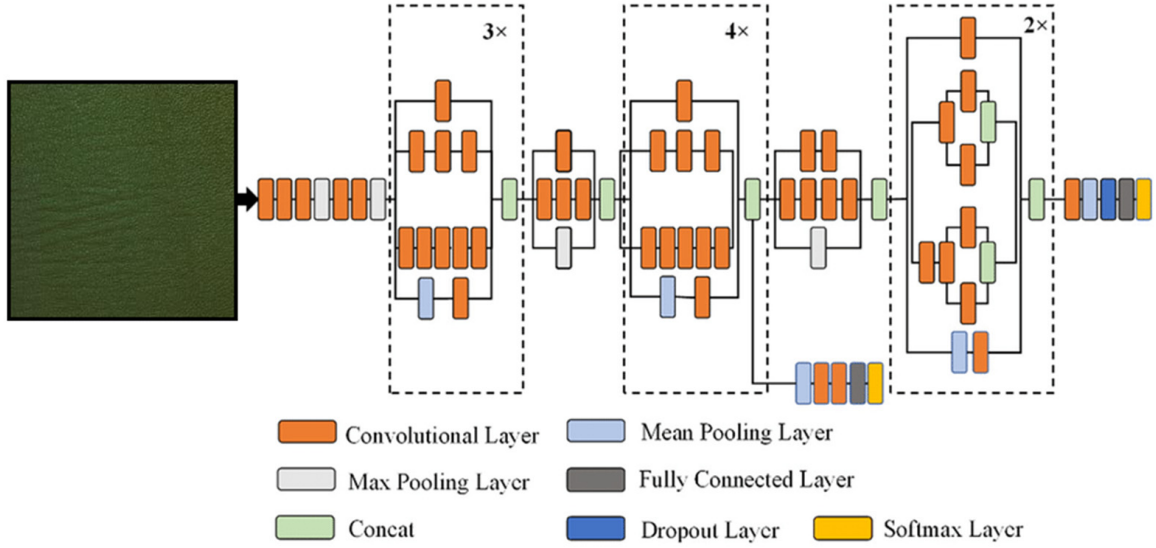



**Figure 2.** A high-level representation of the Inception-V3 architecture.

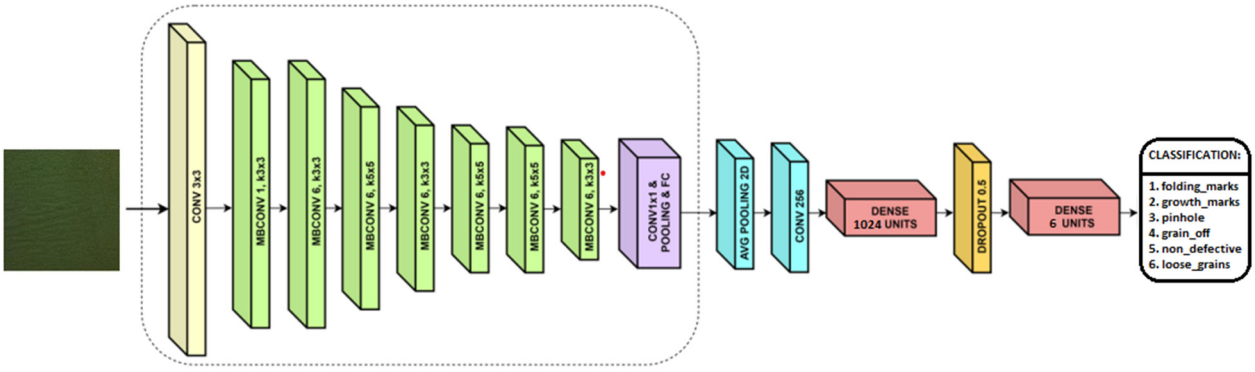



**Figure 3.** The adapted EfficientNet-B0 architecture with one output node for each of the six possible leather defect categories.

## 3. Proposed Method

Vision transformers (ViT) use self-attention mechanisms to process images in a hierarchical manner, capturing both local and global contextual information. The ViT model offers advantages such as scalability, interpretability, and flexibility, making it well-suited for anomaly detection tasks in industries such as manufacturing, healthcare, and security. By training the ViT on large sets of non-defective images, they can learn representations of normal patterns, which can then be used to identify deviations from these patterns as anomalies. These traits allow the ViT to effectively model complex patterns and relationships within images, making them suitable for detecting surface defects in various computer vision-based image domains. Leveraging the model's ability to capture fine-grained details also renders them favourable for identifying subtle anomalies in low-resolution image sets that may be unaccounted for by traditional DL methods.

This section proposes a ViT-based leather surface defect classification method, with the anomalies detected and localised via the ViT architecture.

### *3.1. Dataset Preparation*

The Leather Defect detection and Classification dataset [29] was originally created as a training and validation set for a deep learning neural network-based approach for automated localisation and classification of leather defects using a machine vision system [28]. Although the original dataset was captured in a much higher resolution of 4608 × 3288 pixels, an open-source version was published at a greatly reduced resolution

of 227 × 227 pixels. This dataset with reduced image quality is selected in this study because we focus on achieving defect classification utilising smaller datasets of lower-resolution training images in order to reduce both image-capturing hardware requirements and computational complexity.

In leather manufacturing, colours and finishes are typically introduced through a process known as finishing. The combination of processes such as dyeing, finishing, patination, embossing, fatliquoring, and sealing can result in a wide variety of colours and finishes. Furthermore, while other defects can occur in leather manufacturing, the presence of folding marks, grain off, growth, loose grain, and pinholes in an assortment of colours and finishes provides a comprehensive representation of the most common and impactful defects that arise in the process.

While the colours of sample images can potentially impact classification outcomes, the diversity and distribution of colours across the dataset help alleviate this issue. Furthermore, the vision transformer benefits from processing colour images as opposed to grayscale images due to richer visual cues, such as hue, saturation, and contrast, which can enhance the model's ability to distinguish and classify objects accurately. By incorporating colour information, the vision transformer can potentially capture more fine-grained details and patterns, leading to improved performance.

The dataset comprises a total of 3600 low-resolution sample images, split into 'training' and 'validation' sets, each of which are equally sub-divided into six categories of leather surface images, viz., 'folding marks', 'grain off', 'growth', 'loose grain', 'pinhole', and 'non defective'. Figure 4 illustrates one randomly selected sample image from each category of the dataset.

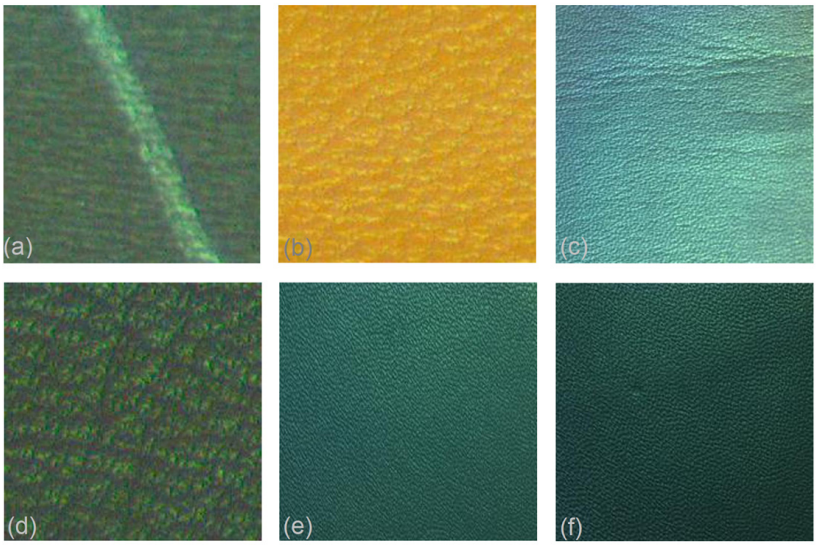



**Figure 4.** Sample images from each class found the leather defect detection and classification dataset [29]: (**a**) folding marks; (**b**) grain off; (**c**) growth marks; (**d**) loose grain; (**e**) no defect; (**f**) pinhole.

The categories in the training set contain 480 unannotated images each, while the categories in the validation set have 120 images, each consisting of a variety of colours and finishes. As the dataset provides a sufficient number (3600) of well-distributed samples across various types, colours, and textures of finished leather surfaces, a good degree of diversity and representativeness is sufficiently captured.

Unfortunately, the dataset contains no clearly annotated defective ground truths as can be found in the popular anomaly detection benchmarking MVTec AD dataset [46]. For this reason, the leather category taken from the MVTec AD dataset is resized to a matching resolution of 224 × 224 pixels in order to confirm the models' ability to identify and localise defects.

### *3.2. Pre-Processing*

From the original leather defect dataset, the validation images for each category are combined with the training images of the same category before test images are completely removed for objective testing later on. The entire dataset is first converted to a NumPy array, reshaped, and pre-processed for sequential data manipulation. The data are then normalized and linked to a 1-hot encoded label list for storage as NumPy files. Validation and training sets are previously combined due to the data and label lists being shuffled and split into training and validation sets after re-loading the NumPy files in each instance, thereby allowing for possible ratiometric adjustments and varied levels of data augmentation without reprocessing the entire dataset.

### *3.3. Proposed ViT Architectures*

The ViT architecture shown in Figure 5 consists of six principal components to be applied to the modified leather defect dataset. Each of these steps are illustrated as follows, using a randomly selected defective sample image as an example.

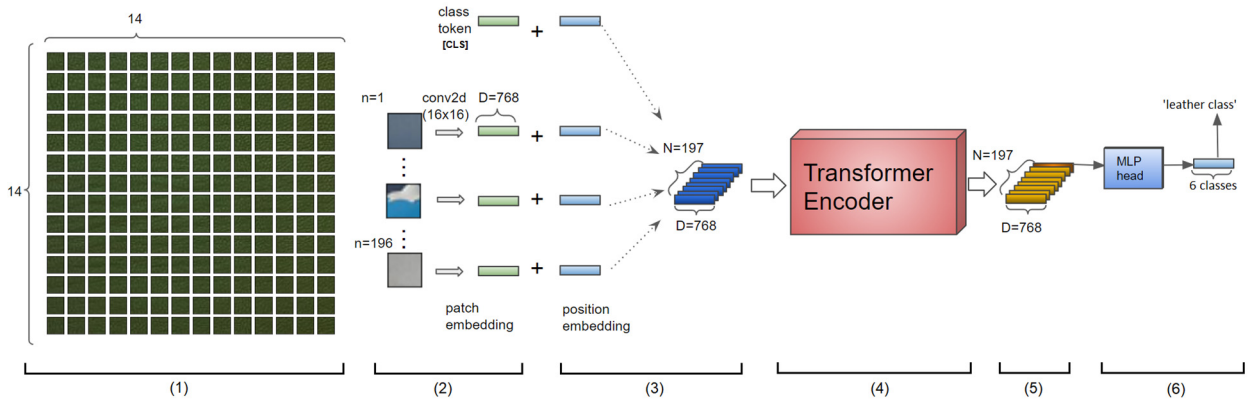
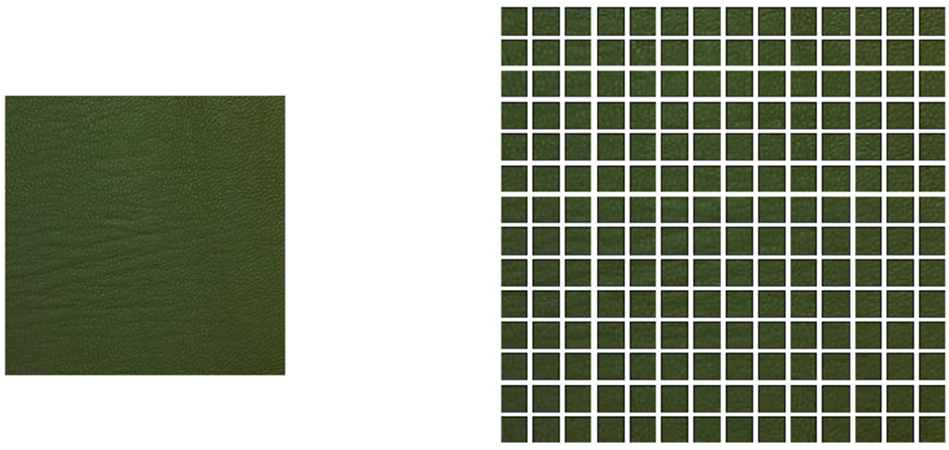



**Figure 5.** Vision transformer inference pipeline (original image [47]).

#### 3.3.1. Patch Extraction

The input image is divided into non-overlapping patches of size $\boldsymbol{P} \times \boldsymbol{P}$, which can be represented as a tensor $\boldsymbol{X}$ of shape $(\boldsymbol{B}, \boldsymbol{H}/\boldsymbol{P}, \boldsymbol{W}/\boldsymbol{P}, \boldsymbol{C})$, where $\boldsymbol{B}$ is the batch size; $\boldsymbol{H}$ and $\boldsymbol{W}$ are the height and width of the image, respectively; and $\boldsymbol{C}$ (=3) is the number of RGB colour channels. Figure 6 depicts a non-defective sample image split into 14 × 14 patches of 16 × 16 pixels each.



(**a**) (**b**)

**Figure 6.** (**a**) Growth-mark defective sample image; (**b**) image split into 14 × 14 patches of 16 × 16 pixels each and converted to D = 768 embedding vectors using a learnable 2D convolution.

3.3.2. Patch Embedding

Each patch is linearly projected into a flat vector by multiplying with a learnable weight matrix $\boldsymbol{E}$ of shape ($\boldsymbol{C}$, $\boldsymbol{D}$), where $\boldsymbol{C}$ is the number of channels and $\boldsymbol{D}$ is the embedding dimension, resulting in a tensor $X_e$ of shape ($\boldsymbol{B}$, $\boldsymbol{H/P}$, $\boldsymbol{W/P}$, $\boldsymbol{D}$).

3.3.3. Positional Encoding

Positional information is added to the patch embeddings to capture spatial relationships. This is typically achieved using sine and cosine functions of different frequencies, denoted as $\boldsymbol{P_E}$, resulting in a tensor of shape ($\boldsymbol{B}$, $\boldsymbol{H/P}$, $\boldsymbol{W/P}$, $\boldsymbol{D}$). To make patches positionally aware, learnable 'position embedding' vectors are added to the patch embedding vectors. As position embedding vectors learn distance within the image, neighbouring vectors have high similarity. Depicted in Figure 7 is the cosine similarity between all-to-all positional embeddings represented as a 14 × 14 grid of similarity maps. Selecting a block ($i$, $j$) from the grid, the colour of each pixel in that block indicates how similar the positional embedding of the patch ($i$, $j$) is to the positional embedding of the patch ($k$, $m$).

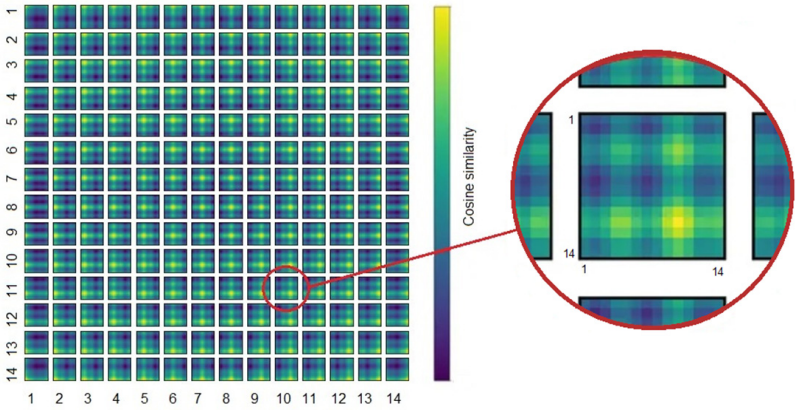



**Figure 7.** Visualization of the cosine similarity between positional patch embedding and all other embeddings based on growth-mark defective sample image.

From the zoomed-in block, it is clear that each positional embedding has a strong similarity to itself as well as those close to it, indicating that the model embeds regions that are physically adjacent with similar vectors. Interestingly, there is more than one bright region, suggesting that various regions use similar positional embedding vectors.

3.3.4. Multi-Head Self-Attention

The patch embeddings with positional encoding are fed into a multi-head self-attention mechanism, where the tensor is linearly projected into query ($\boldsymbol{Q}$), key ($\boldsymbol{K}$), and value ($\boldsymbol{V}$) tensors using learnable weight matrices $\boldsymbol{W_q}$, $\boldsymbol{W_k}$, and $\boldsymbol{W_v}$ of shape ($\boldsymbol{D}$, $\boldsymbol{D}$), respectively. The self-attention mechanism computes the attention scores, $\boldsymbol{A}$, by taking the dot product of $\boldsymbol{Q}$ and $\boldsymbol{K}$, scaled by $1/\sqrt{\boldsymbol{D}}$, followed by Softmax activation, and then multiplied by $\boldsymbol{V}$. The output of self-attention is then concatenated and linearly projected using another weight matrix $\boldsymbol{W_o}$ of shape ($\boldsymbol{D}$, $\boldsymbol{D}$). This process can be visualized, as depicted in Figure 8, where the colour scale illustrates the attention weights' magnitude: blue shades denote low scores, while green shades indicate high scores.

As the attention mechanisms play a role in capturing the relationships and dependencies between different patches, the attention matrices represent the attention weights assigned to each patch or token in a specific head. Each matrix has dimensions equal to the number of patches, while each element within represents the attention weight or relevance assigned to the corresponding row (source patch) and column (target patch) pair.

Figure 9 depicts the input image and 100th rows of attention matrices in heads 0 to 7. These blocks indicate how the 100th patch attends to other patches in the same layer, thereby providing insight into which patches are considered most relevant to the 100th

patch or token in terms of capturing visual features and/or relationships. Each one is a 14-by-14 matrix, while the scale of light to dark patches refers to the distribution of attention weights across the patches.

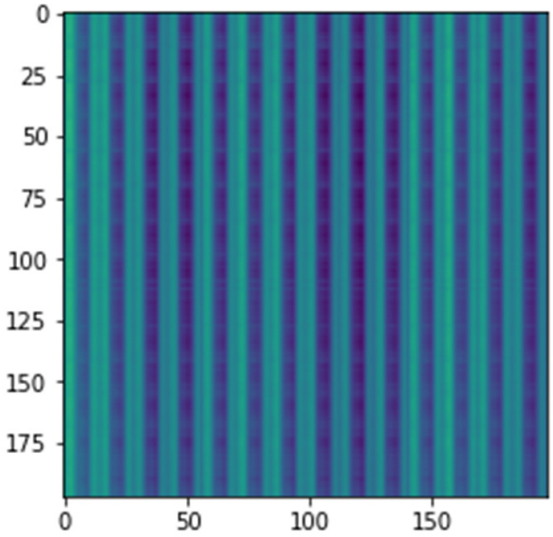



**Figure 8.** Visualization of multi-head self-attention on defective growth-mark sample image.

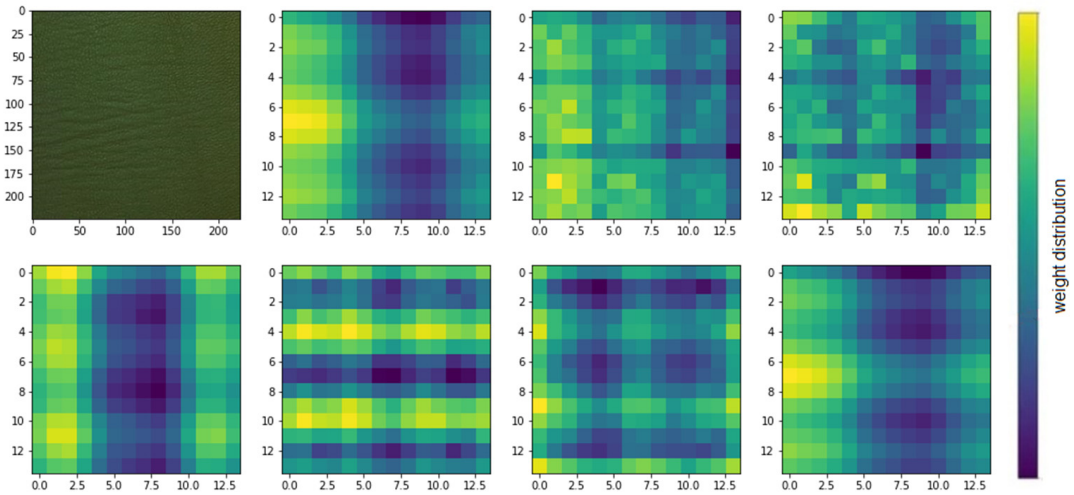



**Figure 9.** Visual representation the 100th rows of attention matrices in the 0–7th heads.

### 3.3.5. Layer Normalization and Residual Connections

After the multi-head self-attention operation, layer normalization is applied to the output to stabilize the features. This is followed by the application of residual connections with skip connections, where the output of the self-attention is combined with its initial input. The combined result is passed through a feed-forward neural network with ReLU activation, enabling non-linearity and further transformation of the features.

### 3.3.6. MLP (Classification) Head

The multilayer perceptron (MLP) [48,49] consists of multiple layers, where each layer transforms its input through an activation function, represented in Equation (1). The MLP typically consists of fully connected layers, where each unit in a layer is connected to all the units in the previous layer and has a unique set of weights.

$$y = \varphi\left(\sum_{j} W_j x_j + b\right) \quad (1)$$

where $x$ is the input vector (output of the preceding layer), $\varphi$ is the activation function, $W$ is the layer weights, and $b$ is the bias vector.

The final representation from the last self-attention block is flattened and passed through a fully connected layer with Softmax activation to obtain class probabilities for image classification. Once constructed and trained, the ViT can be utilised to make predictive classifications on unseen low-resolution leather input images.

### 3.4. Transfer Learning from DL Networks

To compare the training and classification results of the ViT models with the methods obtained from transfer learning from existing pre-trained models, the same dataset is utilised to fine-tune (train) three popular deep learning network architectures. While many larger DL architectures are available, the largest manufacturers of raw leather materials are located in developing or third-world countries. For this reason, consideration must be taken for these developing areas and the financial, hardware, and time constraints involved in the real-world manufacturing environment. To mitigate the requirement for more expensive and powerful hardware while simultaneously reducing model training times, low-end versions of more influential architectures were chosen. The three selected architectures include the ResNet-50 [41], the Inception-V3 [43], and the EfficientNet-B0 [45] networks, pre-trained on the much larger ImageNet dataset [40].

There are two principal methods when applying transfer learning to pre-built architectures. The first is to partially train the model, in which multiple (user selected) top layers from the model are trained on the new dataset, with all layers below being set to frozen. The second method is 'top training', whereby all layers of the model are frozen except the last layer, which is only modified to accommodate the required output tensors.

### 3.5. ViT-Based Anomaly Detection

Each transformer encoder layer consists of a multi-head self-attention mechanism followed by a feed-forward neural network. The self-attention mechanism allows the model to capture global dependencies between different patches or tokens of the input image. After each patch is passed through the stack of transformer encoder-decoder layers, normal and defective image features are refined and represented as feature vectors in high-dimensional space. The final output of the ViT model is a sequence of feature vectors, where each vector corresponds to a specific position or patch in the input image. Using the features extracted from intermediate layers of the trained vision transformer models, normal and defective regions are predicted in the test dataset to produce prediction weights.

In this paper, two ViT-based anomaly detection methods are considered.

The first ViT-based anomaly detection method is to calculate the dot product similarity between the extracted feature vectors and the feature vectors of input image, as well as the predicted class of the input image. By reshaping the calculated similarity scores to original image dimensions, it is possible to generate a heatmap that highlights anomalous data regions with pixel-level localisation of defects. The sensitivity of the classification outputs can be adjusted by tuning the anomaly threshold value to meet specified industry requirements.

The second ViT-based anomaly method is the modified AnoViT [50], whereby the distribution of the normal data is learned from the differences between the ViT-based encoder–decoder image reconstruction and the original input image. Pixel-level $l_2$-distances between the two are calculated, as well as the average pooling across the channels to produce a score map. As an unsupervised learning method, the model is trained on normal data only and is therefore only able to learn the distribution of non-defective images. This means that abnormal or defective images will cause an increase in reconstruction error, as anomalous regions will be reconstructed incorrectly. By integrating the ViT-based encoder into an encoder–decoder architecture, the AnoViT model [50] leverages core vision transformer characteristics to achieve an improved method of anomaly detection and localisation. An architectural diagram of the AnoViT model is shown in Figure 10,

where $E_{[n]}$ denotes embedded patches after linear projection, $E^n_{pos}$ are positionally encoded patches, and $E'_{[n]}$ are the output patch embeddings.

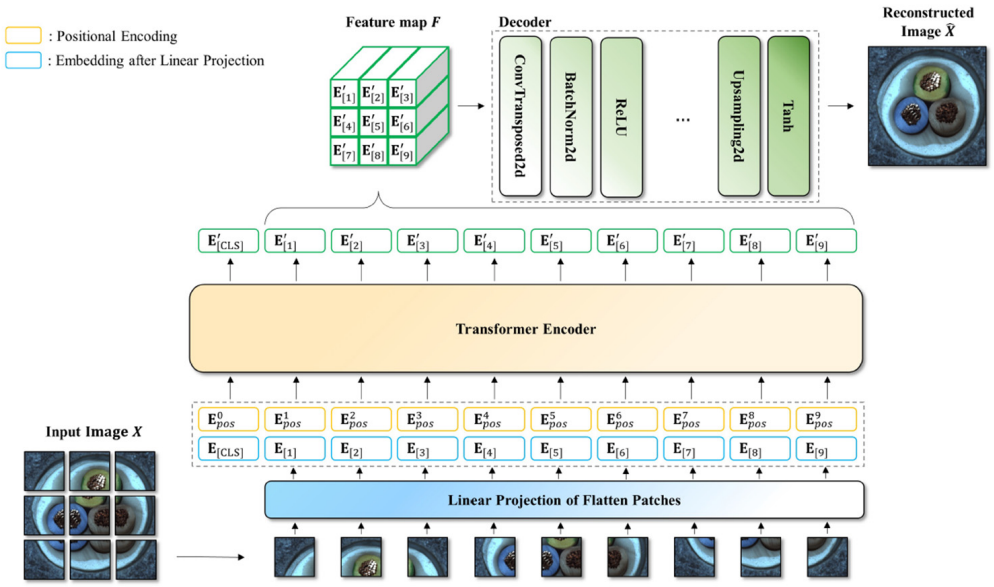



**Figure 10.** Architectural diagram of the AnoViT model [50].

By applying a multi-head self-attention (MSA), the relationship between patches is learned in order to utilise global information. Additionally, by processing patch-level images, the approach creates image embeddings with rich information for each location.

Typical ViT models consider image patches of a specific size as tokens in the NLP tasks. When performing classification, the 'CLS' token is added at the beginning of the resulting sequence:

$$[x_{class}], \left[x_p^1, \ldots, x_p^N\right], \tag{2}$$

where $x_p^i$ are image patches. Where the architecture has multiple layers, the state of the CLS token on the output layer is used for classification. Unlike these models, the AnoViT excludes the patch embedding's $E_{[cls]}$ token and additional MLP head, then creates a feature map by rearranging the remaining embeddings to match the existing positions in the patch image and used as the feature map $\Gamma$.

From the reconstruction error, the activation map is generated for each input image depicting anomalous regions of interest, such as surface defects.

Having been trained on normal data, the model can learn the distribution of normal data only. Owing to this, an atypical input image causes reconstruction error to increase as the model has difficulty reproducing anomalous regions. To localise areas containing anomalies in a new input image, the image reconstructed at the output of the ViT-based encoder–decoder model $f$ is used to calculate the $l_2$-distance between the two on a per-pixel basis. The score map $M$ is calculated by taking the average pooling across the channels (Equation (3)).

$$M = \|\mathrm{X}_{ij} - f\left(\mathrm{X}_{ij}\right)\|_2, \tag{3}$$

$$s_a = \max M. \tag{4}$$

The anomaly score is calculated by extracting the highest value from the score map (Equation (4)). Anomaly localisation is then achieved by utilising the score map $M$ to evaluate the abnormality of each pixel in the image.

### 3.6. Evaluation

While adhering to the described vision transformer architectural structure, two variations of the ViT model are tested on the same modified low-resolution leather defect dataset for a comparative evaluation of performance metrics and classification results.

As high attention values can be interpreted as the model finding those regions more important for the task at hand, and therefore by visualizing the "attention" as a heatmap over the image, attention mechanisms provide some model interpretability. However, while this allows a rough sense of what the model is "looking at", the attention mechanism is only one component of the model's decision-making process and does not fully explain the model's decision-making process. Despite the ViT delivering high performance in computer vision tasks, it presents interpretability challenges due to its complex architecture. Unlike convolutional neural networks (CNNs), which process local, contiguous information, the ViT handles images as sequences of patches, applying self-attention mechanisms and feed-forward neural networks. This non-local processing makes understanding feature importance and tracing decision-making pathways difficult. Existing interpretability methods like attention rollouts, feature attribution, and saliency maps, while promising, need further development to effectively untangle the ViT model's decision-making processes.

#### 3.6.1. Evaluation

Evaluation of the models' training and validation performances is conducted using typical metrics, such as accuracy, loss, precision, recall, F1-score, receiver operating characteristic (ROC) curves, and the area under this curve (AUC).

#### 3.6.2. Fine-Tuning

Fine-tuning is achieved via experimentation (trial and error) with varying hyperparameter values throughout the training stages to optimize model performance.

#### 3.6.3. Validation

The models' performance is validated by averaging the classification results of multiple novel input images from each leather category, as well as the visual analysis and confirmation of activation maps for localised anomalous regions. Classification results are then compared to those derived from the application of transfer learning of popular deep learning classifiers (ResNet-50, Inception-V3, and EfficientNet-B0).

## 4. Experiments

In order to evaluate the influence of the attention mechanism, tokenization, and hyperparameters within the vision transformer model, two slightly varied implementations of the model are modified and trained on the same low-resolution leather defect dataset. Metrics and results are then compared between the two ViT variations and three state-of-the-art deep learning models (ResNet-50, Inception-V3, and EfficientNet-B0).

### 4.1. Setup

#### 4.1.1. Dataset

Prior to pre-processing, ten defective images from each of the six categories (60 images) were completely removed from the reduced resolution Leather Defect detection and Classification dataset [29] for objective classification after training. The remaining 3540 images (227 $\times$ 227 pixels) in all six categories are resized to 224 $\times$ 224 pixels and converted to a NumPy array before adding a new axis so that the array has a shape of (1, height, width, channels). Images are then pre-processed and sequentially added to a 'data list'. Once all images are complete, the entire data list is converted to a NumPy array, the data type is altered, and the pixel values are normalized to the range of 0–1. The new shape of the NumPy array is (3540, 1, 224, 224, 3). Finally, the first two elements are swapped, and the first element is removed from the array, so that the final 'data array' shape is (3540, 224, 224, 3). A new NumPy 'label array' is then created corresponding to the size of the data array

and one-hot encoded before both arrays are saved as NumPy files for consistent testing with multiple models.

After loading the NumPy files, the data and labels are shuffled and split into training and validation sets with the ratio 0.8 to 0.2 (2832 training to 708 validation). Before training, the dataset is augmented using techniques including rotation, flipping, and scaling to increase the diversity of the data and improve model generalization.

#### 4.1.2. Model Variants

The first configuration of the ViT model utilises the conventional vision transformer structure, as described in Section 3.3, with patch extraction and standard multi-head attention. It includes an MLP head layer [48] with a Softmax activation function for multi-class image classification.

The second ViT model differs in the following three ways. It employs a shifted patch tokenization layer for patch extraction, with a custom multi-head attention layer which uses a linearly scaled dot product attention mechanism with a diagonal attention mask. This attention mechanism is designed to have a lower computational complexity than the original multi-head attention. The model also includes the MLP head layer; however, instead of a Softmax activation, it returns the logits directly.

The anomaly detection method is a ViT-based encoder–decoder architecture modified from the AnoViT [50], which is an unsupervised method that utilises extracted patch embeddings to reflect the global context of the image from the attention-based ViT encoder. With the embeddings of each image patch, reconstruction error at the image and pixel levels is calculated to both detect and localise image anomalies by deriving the reconstructed image's activation map at the output.

Comparative supervised learning methods include three transfer learning architectures, ResNet-50, Inception-V3, and EfficientNet-B0. Modifying the original ResNet-50 architecture requires adjusting the input and output shape to [224, 224, 3], to match dataset dimensions, and the last pooling layer before the 1000 class dense layer to be extracted, so that the output can be flattened for a new dense layer to be appended. Although the new layer still uses the Softmax activation function, it is modified from the 1000 output nodes or classes to accommodate only six classes.

As with the ResNet-50 model, modifying the Inception-V3 model incorporates freezing all layers except the last layer, before creating a new input layer with matching dimensions to those of the reshaped input data. The extracted output from the max pooling layer (last layer before the output layer) is flattened and linked to a newly constructed dense output layer with a Softmax activation function, where the original 1000 output nodes were redefined to six nodes or classes.

The EfficientNet-B0 architecture is altered by extracting the final pooling layer before the dense layer containing 1000 classes. From here, however, a slightly different approach from the previous two models is taken. Instead of flattening the output layer, a global average pooling is applied to the data instead. This technique leaves other dimensions untouched, while applying average pooling to the spatial dimensions until each spatial dimension is reduced to one. Unlike the process of flattening, the values are not retained due to being averaged; however, global spatial information is captured by summarizing the feature maps across the entire spatial extent of the image. From here, a dense layer is added with an output of 1024 and the rectified linear units (ReLU) activation function. The final layer is another dense layer, this time with an output shape of (none, 6) to match the number of categories in the dataset and a 'softmax' activation function for multi-class classification. To significantly decrease training time of the EfficientNet-B0 model, the "efficientnet_b0_weights.h5" weights file is utilised in the 'ModelCheckpoint' callback. The modified model is depicted in Figure 3.

Before training the modified models, the 'Early Stopping' callback is included, which monitors the validation loss metric and ends the training session when no significant improvement to validation loss is detected over a predefined 'patience' or consecutive

number of epochs. After training each model, the 10 unseen test images taken from each category (60 images total) are input for classifier prediction.

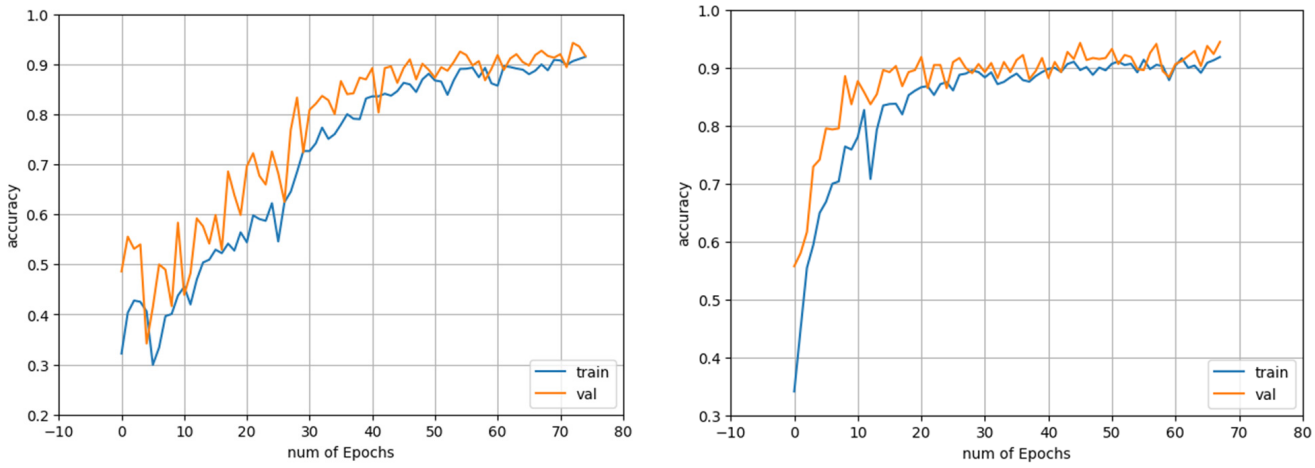
For the sake of consistency throughout training stages, all DN and ViT models were compiled and fit with the 'categorical cross-entropy' loss function and the 'adam' optimizer, and for probability distributions in multi-class classification, the 'softmax' activation function. Other optimizers and loss functions were also tested with each model; however, it was found that this combination yielded optimal results overall.

#### 4.1.3. Fine-Tuning

Though more specific adjustments are stated in the specific model implementations, some general fine-tuning techniques can be applied across all models. Proper selection of learning rate assists in efficient model convergence and improved accuracy, while an optimal weight decay can improve the generalization ability and prevent overfitting. A higher upper limit of epochs can be selected, as well as an increased 'patience' parameter for the 'early stopping' callback method. Batch normalization for improved training efficiency and consistency. Increasing the batch size, within hardware limitations (CPU and GPU memory availability), can positively influence training speed, generalization, and learning dynamic. Data augmentation and dropout for model robustness and prevention of overfitting. Further training optimization is achieved with adjustments to the number of parallel processes ('num_workers') and automatic mixed precision training ('amp') parameters. Finally, experimentation with ViT-specific hyper-parameters such as the patch size, projection dimension, number of heads, and transformer layers can influence the model's ability to capture spatial information and patterns in the input images.

### *4.2. Performance Metrics*

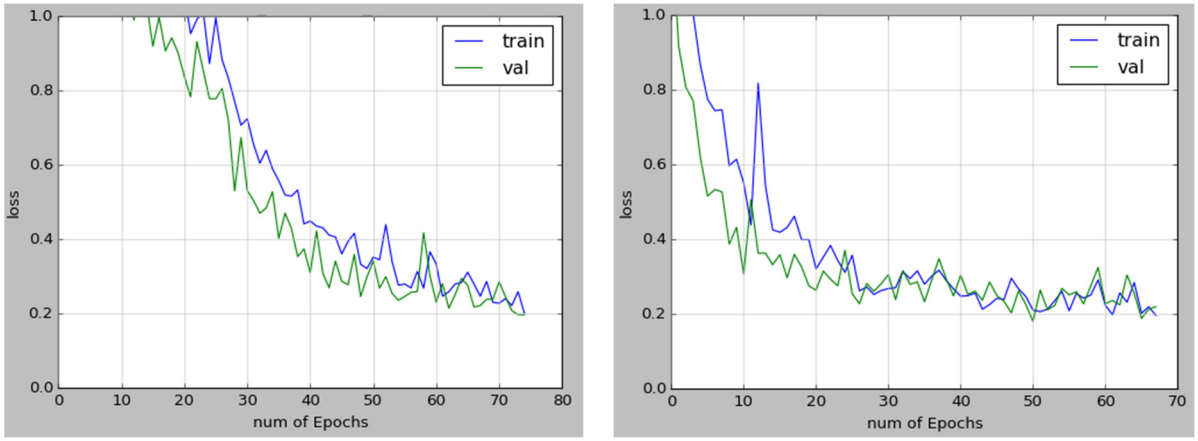
When comparing the two ViT models trained on the low-resolution leather dataset, it is evident that both models displayed similar training and validation accuracy ROC curves. However, by optimizing training efficiency with the 'early stopping' callback method and a patience of 10, there is a notable difference in their accuracy plots (shown in Figure 11). The second ViT model (ViT-02) achieves a significantly higher validation accuracy of 93.89% in just 68 epochs compared to the first (ViT-01), with an average validation accuracy of 89.86% after stopping at 75 epochs. This indicates that ViT-02 performed better in terms of validation accuracy, suggesting improved classification or prediction capability compared to ViT-01 within a shorter training duration.




(**a**) (**b**)

**Figure 11.** Receiver operating characteristic (ROC) curves for training accuracy versus validation accuracy: (**a**) ViT-01; (**b**) ViT-02.

In Figure 12, the validation loss comparison between the two ViT network configurations indicates a similar result; however, ViT-02 reaches a lower average validation loss of 0.1846 after stopping at 68 epochs compared to ViT-01's average validation loss of 0.1970 at 75 epochs. This indicates that ViT-02 achieved an improved accuracy or generalization compared to ViT-01 within a shorter training duration. Also noted in each of these figures is that ViT-02 has a much steeper gradient and therefore reaches its optimal range more efficiently than ViT-01.




(**a**) (**b**)

**Figure 12.** Receiver operating characteristic curves for training loss versus validation loss: (**a**) ViT-01; (**b**) ViT-02.

Figure 13 depicts the heatmap confusion matrices for the categorical classification performance during training of each vision transformer. It is evident that although trained on a small number of low-resolution images, both models perform similarly, with a relatively high number of true positives (TP) in each category of the leather dataset.

It can also be seen in Figure 14 that by comparing the average precision, recall, and F1-scores for each leather category, there is relatively little difference between the two models' training performances; however, ViT-02 performs equally well as or outperforms the original ViT model in all leather categories.

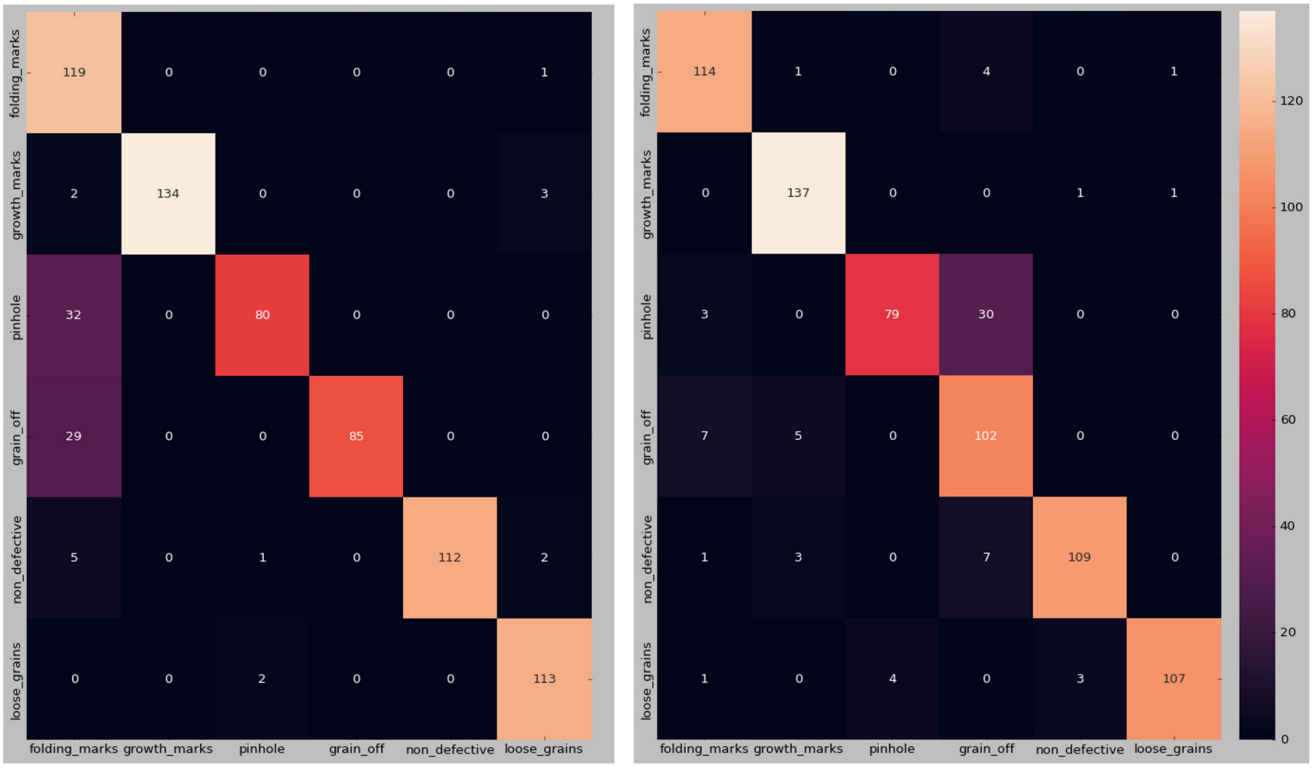
Table 1 displays the average training metrics of each tested architecture trained on the Leather Defect detection and Classification dataset. From this comparative analysis, it is evident that the ResNet-50 model exhibited subpar performance and also takes the longest time to train, further emphasizing its inefficiency. Conversely, the EfficientNet-B0 and Inception-V3 models demonstrated superior results in terms of average validation accuracy and precision when compared to ViT-01. Between Inception-V3 and EfficientNet-B0, the former shows a slightly higher precision but a slightly lower recall than the latter. Inception-V3 not only has a higher AUC, indicating that it performs slightly better across various classification thresholds, but it also takes significantly less time to train than EfficientNet-B0, making it the more efficient model between the two. Of the five models in Table 1, ViT-02 demonstrated superior overall performance and generalization ability reached in almost half the training time of ViT-01, making it a more robust and effective model for the categorical classification of leather surface defects. Additionally, ViT-02 has the shortest training time (176.27 s), suggesting that it is the most computationally efficient model.

**Table 1.** Training time and average performance metrics for the ResNet-50, Inception-V3, EfficientNet-B0, ViT-01, and ViT-02 models trained on the low-resolution leather dataset.

| Model | Accuracy | Loss | Precision | Recall | F1-Score | AUC | Training Time |
|---|---|---|---|---|---|---|---|
| ResNet-50 | 0.3983 | 1.4732 | 0.6806 | 0.0169 | 0.0328 | 0.7790 | 769.69 s |
| Inception-V3 | 0.9152 | 0.2347 | 0.9248 | 0.8737 | 0.8984 | 0.9944 | 292.34 s |

**Table 1.** *Cont.*

| Model | Accuracy | Loss | Precision | Recall | F1-Score | AUC | Training Time |
|---|---|---|---|---|---|---|---|
| EfficientNet-B0 | 0.9102 | 0.2814 | 0.9213 | 0.8980 | 0.9090 | 0.9907 | 572.89 s |
| ViT-01 | 0.8986 | 0.1970 | 0.9131 | **0.8981** | 0.9054 | 0.9944 | 341.01 s |
| ViT-02 | **0.9389** | **0.1846** | **0.9609** | 0.8958 | **0.9271** | **0.9963** | **176.27 s** |




(**a**) (**b**)

**Figure 13.** Heatmap confusion matrix for the classification accuracy of the two models on each of the six leather categories: (**a**) ViT-01; (**b**) ViT-02.

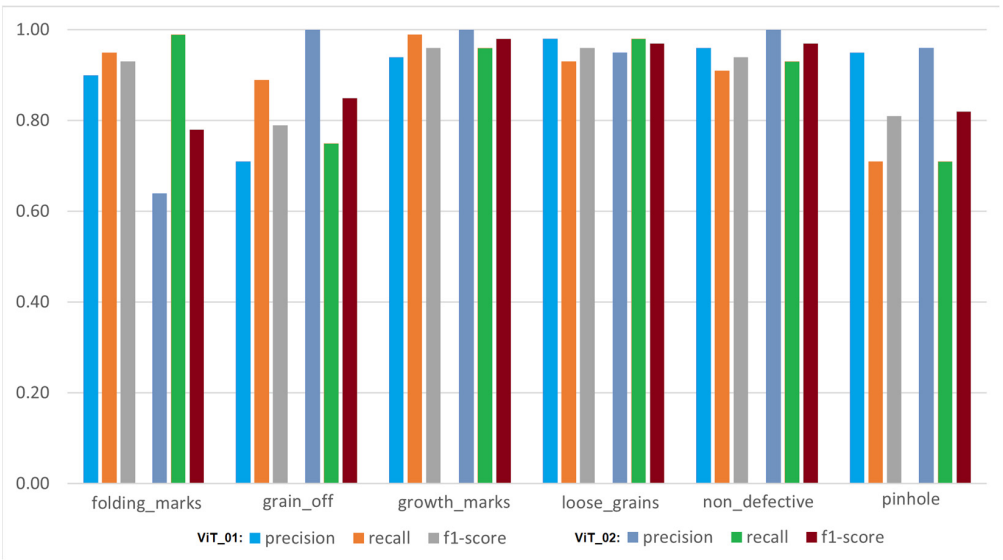



**Figure 14.** Comparison of average performance metrics for ViT-01 and ViT-02; precision, recall, and F1-score for each of the six leather categories.

### 4.3. Results and Discussion

#### 4.3.1. Defect Classification Accuracy

By completely removing 10 images from each category (60 images) before data preprocessing and training of the models, an objective classification test of unseen input images yielded the results shown in Table 2. Owing to the low-resolution and small number of leather surface images in the training set, relatively low categorical classification accuracies (six-class classification) are to be expected. ViT-02 outperforms all other methods.

**Table 2.** A comparative table between two ViT network configurations and three transfer learning CNN classifiers; columns depict the average correctly predicted classification scores over all six leather categories as well as defective and non-defective classes.

| | ViT-01 | ViT-02 | ResNet-50 | Inception-V3 | EfficientNet-B0 |
|---|---|---|---|---|---|
| Inference time (approx. per image) | 104.52 ms | 79.48 ms | 108.15 ms | 107.47 ms | **63.24 ms** |
| Avrg 6-Class Acc. | 36.67% | **53.33%** | 36.67% | 45% | 48.83% |
| Avrg 2-Class Acc. | 76.67% | **86.67%** | 71.67% | 83.33% | 83.33% |

In terms of each vision transformer predicting whether an input image is defective or non-defective (the two classes of defective or not), ViT-01 is outperformed by the Inception-V3 and EfficientNet-B0 models, while ViT-02 outperforms all models. However, when predicting one of the six specific leather categories, the results show that ViT-02 outperforms the EfficientNet-B0 model by two correct predictions. The average approximate inference time per image is calculated by dividing the time taken for all 60 unseen images by 60. From this, we can see that while ViT-01 is slightly faster than the ResNet-50 and Inception-v3 models, the EfficientNet-B0 and ViT-02 are significantly faster than these three. Although the ViT-02 model is slower than the EfficientNet-B0 model by approximately 16.24 milliseconds per image, it is redeemed by being more accurate in both binary and multi-class classification.

#### 4.3.2. Dot Product (Similarity) Anomaly Detection and Localisation

Training each of the two ViT models on 600 normal samples and 600 combined defective samples (120 random images from each of the five categories), not only can defects be detected but also localised. By extracting activations from intermediate layers, normal and defective features can be represented as vectors, where each image corresponds to a feature vector in a high-dimensional space. The classification of new input images as normal or defective can be found by calculating the similarity (dot product or cosine similarity) between these vectors and the vector of a new input. Similarly, the dot product can be calculated between each input patch feature vector and the corresponding normal image patch feature vector. A significantly low similarity suggests the patch is anomalous. Having been trained to recognize specific defects, a heatmap can be generated that can localise abnormal regions at the pixel level. A high-level anomaly heatmap can be generated where a patch's colour intensity is inversely proportional to its similarity score with normal images. By visualizing the activation maps of the network's layers, high activations typically correspond to the regions where the model has detected certain features that correspond to surface defects, as shown in Figure 15.

Using a threshold value, "peaks" or hot spots are detected and red bounding boxes are drawn around these areas to further aid in defect classification analysis. Adjusting the anomaly threshold value can fine-tune the sensitivity of the classification outputs to meet industry requirements. Other useful heatmap overlay parameter adjustments include reducing overlay saturation with a pixel value off-set, as well as setting heatmap values in the dark regions to 0 as the edges of dark pixels may be thought of as high anomaly.

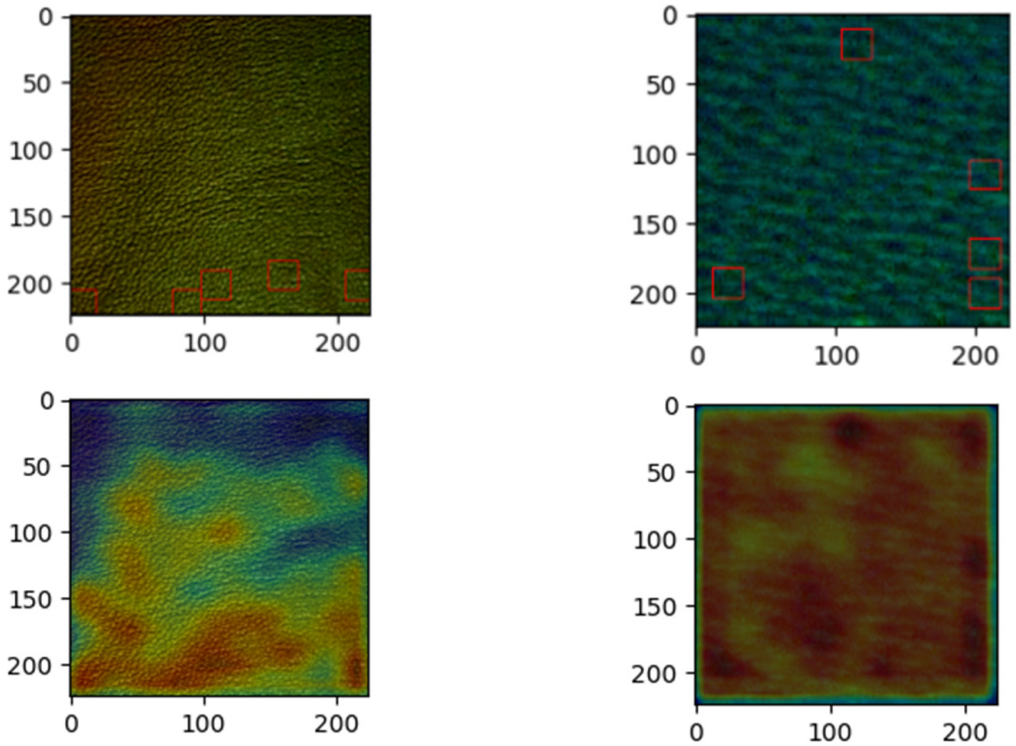



**Figure 15.** ViT-based dot product anomaly detection and localisation: randomly selected validation image (**top**) and corresponding activation map overlay with localised defective regions of interest (**bottom**). (**a**) Grain off defect; (**b**) Loose grain defect.

#### 4.3.3. AnoViT Anomaly Detection

To train the vision transformer-based encoder-decoder model (AnoViT) on the low-resolution Leather Defect detection and Classification dataset, several modifications are required. As an unsupervised learning method, only normal images are required for training. After the training and validation datasets are split in the ratio of 0.8 to 0.2, the remaining 472 non-defective leather sample images are utilised for training the model. Several configuration parameters must be adjusted for the reduced image size (from the original scripts input dimensions of 1024 $\times$ 1024 pixels), as well as the increased number of input images, and the unavailability of validating ground truth images.

Figure 16 depicts the original input image (top) and its corresponding class activation map (bottom) generated using the reconstruction error based on learned distribution of normal images.

Figure 17 shows that visibly prominent defects found in the benchmark MVTec AD leather dataset [46] are identified relatively clearer. However, attributed to the fact that the 'normal' data used to train the model possess no defects and therefore no shadows, some reconstruction error is found in any abnormalities, including slight contours and shadows caused by defects such as the cut and poke defects. To better match the required ratiometric output dimensions as well as patch size divisions, images are processed and exported at decreased dimensions of 512 $\times$ 512 pixels (half the original size).

Aside from the primary defect identified in box 2 of both images, it is worth noting a line at the top (shown in box 1) of many reconstructed output images caused by an error in the reconstruction process, though the line itself is not indicative of any defects in the input image. This reconstructive error is found in both datasets (seen in Figure 16a) but is more prominent in the images from the MVTec AD dataset.

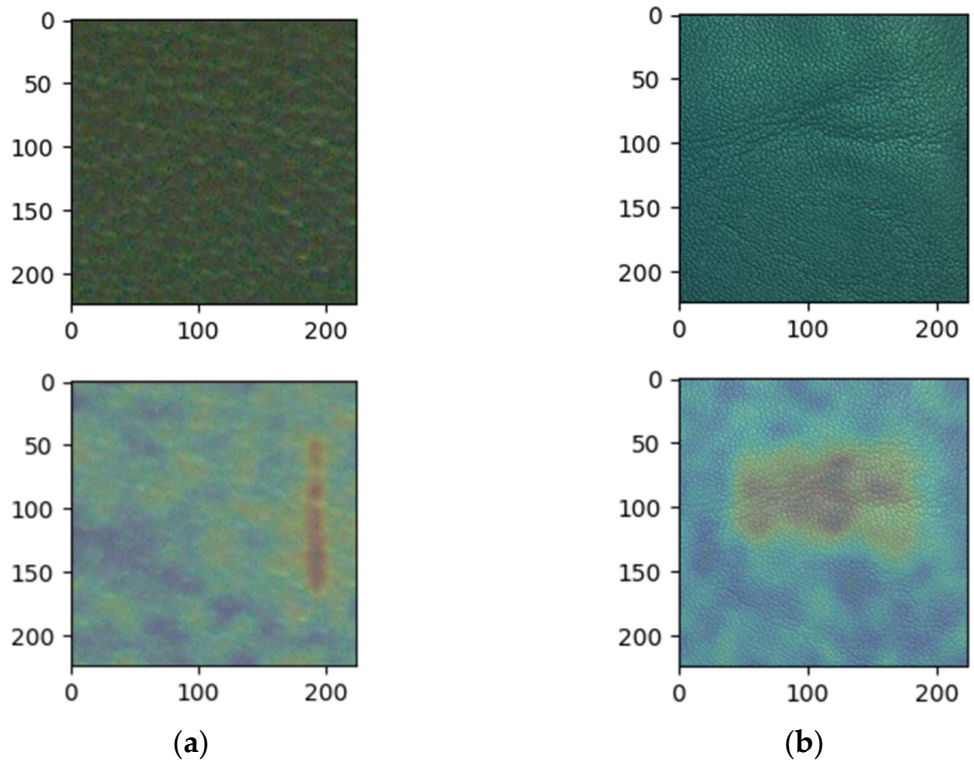



**Figure 16.** Vision transformer-based encoder–decoder anomaly detection (AnoViT): randomly selected low-resolution validation image (**top**) with corresponding activation map overlay (**bottom**); Image scale in pixels. (**a**) Loose grain defect; (**b**) Folding mark defect.

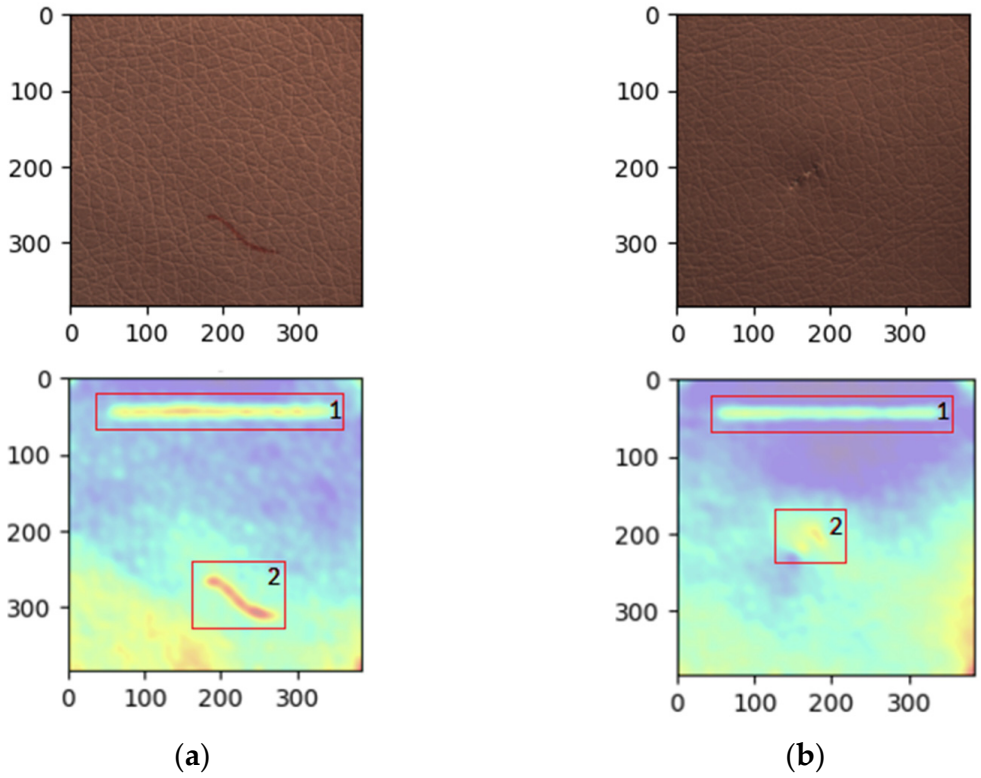



**Figure 17.** Vision transformer-based encoder–decoder anomaly detection (AnoViT): randomly selected validation image (**top**) and corresponding activation map overlay with localised defective regions of interest (**bottom**); Image scale in pixels. (**a**) Colour defect; (**b**) Cut defect image.

To enhance the performance on low-resolution images, techniques like histogram equalization and contrast enhancement can be employed, improving the quality of low-resolution images before they are fed into the model. Model performance is also refined by better learning generalized features with increased data augmentation techniques such as rotation, flipping, zooming, translation, and brightness. To further improve the sensitivity and performance of the AnoViT anomaly detection method, it was found that fine-tuning the parameter constants, within their provided ranges, produced improved results. Sensitivity to certain surface or defect types is increased by manually experimenting with various combinations of hyper-parameter values. Variations in static array values (mean and std) used for image denormalization, optimizer parameters (learning rate, beta1, and beta2), and ViT-specific characteristics (patch and batch size, number of epochs, weight decay, and validation ratio) resulted in observable result improvements in certain defect types. Although experimentation with adjusted hyper-parameter values did produce better localisation results, improvements were not necessarily achieved across all categories (indistinguishable change in pinhole defects). Some simple post-processing techniques, such as image brightness and contrast, also enhanced defect visibility in all output results. Refined localisation results can be seen in defective categories with more prominent surface defects, as seen in the folding mark and loose grain sample images in Figure 18.

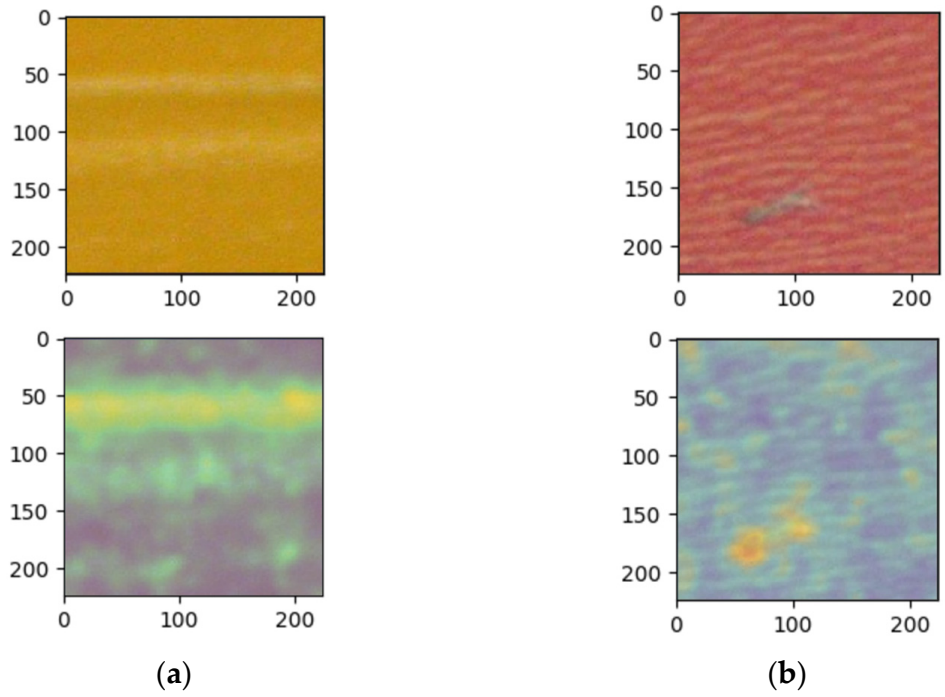
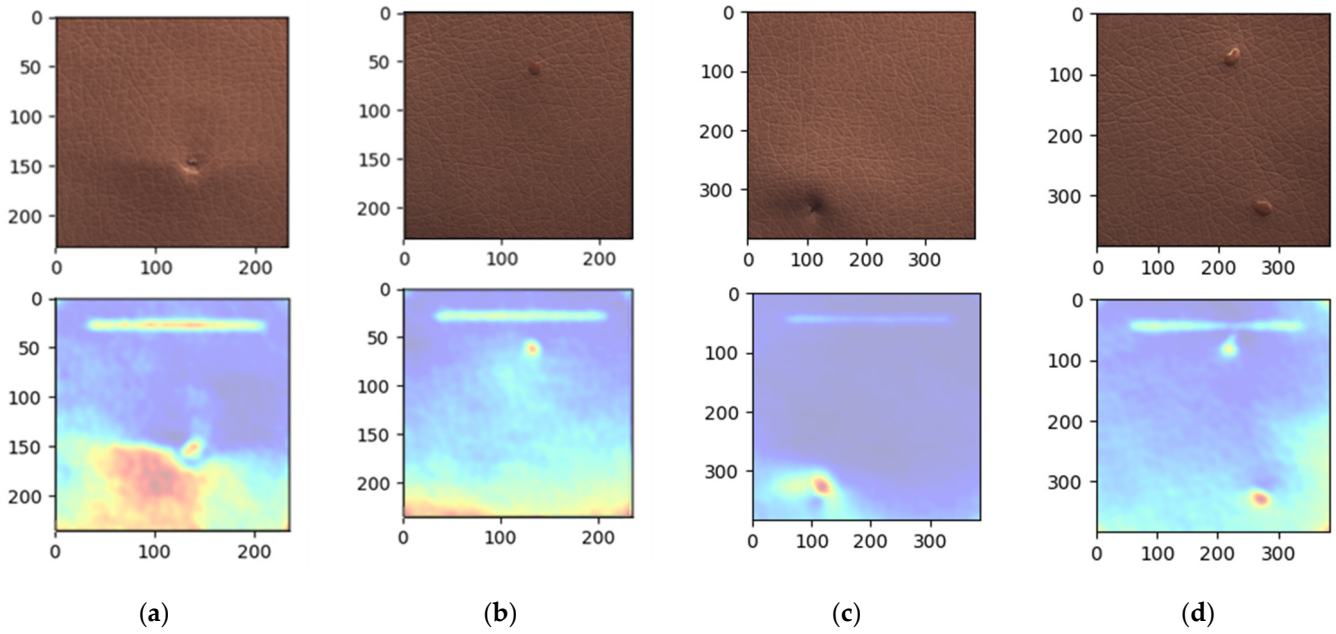



**Figure 18.** Vision transformer-based encoder–decoder anomaly detection (AnoViT) fine-tuned results: randomly selected validation image (**top**) and corresponding activation map overlay with localised defective regions of interest (**bottom**). (**a**) Folding mark defect; (**b**) Loose grain defect.

### *4.4. Comparison with State-of-the-Art Transformer-Based Methods*

This section provides a comparative analysis of two vision transformer (ViT) models, focusing on their performance in detecting and localising anomalies as well as classifying defects found in images of leather surfaces.

When applying the AnoViT method and reducing the resolution of the MVTec AD leather dataset by a factor of 4.57, i.e., from 1024 × 1024 pixels reduced to 224 × 224 pixels, not only is the reconstruction error increased due to noise, but the false reconstruction error (not indicating defective region) becomes more prominent. The difference in defect localisation between these resolutions is illustrated in Figure 19.



(**a**) (**b**) (**c**) (**d**)

**Figure 19.** ViT-based encoder–decoder anomaly detection (AnoViT) tested on defective MVTec AD leather sample images. Input image (**top**) and corresponding activation map with localised defective regions (**bottom**): (**a**) poke defect (224 × 224 pixels); (**b**) glue defect (224 × 224 pixels); (**c**) poke defect (1024 × 1024 pixels); (**d**) glue defect (1024 × 1024 pixels).

Owing to more visibly prominent defects found in the MVTec AD dataset, it can be noted that the falsely identified anomalous regions are more pronounced in lower-resolution images (224 × 224 pixels) when compared to higher-resolution images (1024 × 1024 pixels). The error intensity can, however, be reduced in low-resolution images with further fine-tuning of hyper-parameters for the datasets with dissimilar lighting and surface types.

A comparison of the ViT-01 model's validation accuracy of 89.86% reached after a training time of 341.01 s and the ViT-02's validation accuracy of 93.89% reached after a training time of 176.27 s conveys a significant improvement in accuracy (4.03%) achieved in nearly half the training time (Table 1). The fact that the ViT-02 model is fully trained in 2 min and 56.27 s on a standard laptop (Intel core i7 with 6 Gb GPU) suggests that the classifier model can be quickly re-trained on a new or modified dataset and deployed in a real-time environment where efficient decision making is essential. Once trained, the model achieved inference of all 60 unseen test images in 4.76 s (79.3 milliseconds per image), making it a viable real-time classification model.

When comparing the anomaly localisation results produced by the AnoViT and the Dot Product methods on low-resolution images, it can be seen that the latter method did perform acceptably over the majority of the defect types found in the six leather categories; however, the former method localised multiple defects in a single image with more clarity. While the AnoViT method requires a moderate increase in computational resources, the results produced suggest it is a more robust and effective model for the detection and localisation of leather surface defects in low-resolution images.

## 5. Conclusions

This study modified the ViT architecture for anomaly detection and localisation and defect classification based on low-resolution leather surface images and small-size datasets. By harnessing the self-attention mechanisms inherent in ViT, the model is able to process images in a hierarchical manner, effectively capturing both local and global contextual information. This capability empowers the model to accurately model intricate patterns and relationships within images, making it a suitable choice for detecting anomalies. Through

experimentation and analysis conducted on the publicly available Leather Defect Detection and Classification dataset [29], and the anomaly benchmark dataset MVTec AD [46], the ViT model has demonstrated its efficacy in visualizing regions of interest that deviate from the norm and accurately classifying input images as defective or non-defective. Based on the results obtained using the vision transformer model, the system not only assisted the trained technicians in leather grading by pre-classifying the defect category as well as highlighting defective regions of interest using heatmaps and bounding boxes, but also contributes to the design of an updated, fully automated leather grading system.

**Author Contributions:** A.D.S. algorithm development and implementation, experiments, and paper writing. S.D. and A.K. supervising and paper revision. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** All tested models are available at GitHub. Datasets are available at Kaggle and MVTec AD

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Fortune Business Insights. Leather Goods Market Size, Share & COVID-19 Impact Analysis, by Source (Full Grain Leather and Synthetic Leather), by Product (Apparel, Luggage, Footwear, and Others), By End-user (Men, Women, and Kids), and Regional Forecast, 2023–2030. In *Market Research Report*; FBI104405; Fortune Business Insights: Pune, India, 2023.
2. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
3. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2012**, *60*, 84–90. [CrossRef]
4. Sermanet, P.; Eigen, D.; Zhang, X.; Mathieu, M.; Fergus, R.; LeCun, Y. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *arXiv* **2013**, arXiv:1312.6229.
5. Abdulateef, S.K.; Salman, M.D. A Comprehensive Review of Image Segmentation Techniques. *Iraqi J. Electr. Electron. Eng.* **2021**, *17*, 166–175. [CrossRef]
6. Vaswani, A.; Shazeer, N.M.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. *arXiv* **2017**, arXiv:1706.03762.
7. Kendall, E.J.; Barnett, M.G.; Chytyk-Praznik, K. Automatic detection of anomalies in screening mammograms. *BMC Med. Imaging* **2013**, *13*, 43. [CrossRef]
8. Gyimah, N.K.; Girma, A.; Mahmoud, M.N.; Nateghi, S.; Homaifar, A.; Opoku, D. A Robust Completed Local Binary Pattern (RCLBP) for Surface Defect Detection. In Proceedings of the 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Melbourne, Australia, 17–20 October 2021; pp. 1927–1934.
9. Casanova, E.Z.; García-Bermejo, J.G.; Medina, R.; Fernández, J.L. Road Crack Detection Using Visual Features Extracted by Gabor Filters. *Comput.-Aided Civ. Infrastruct. Eng.* **2014**, *29*, 342–358.
10. Vaideliene, G.; Valantinas, J. Wavelet-based Defect Detection System for Grey-level Texture Images. In Proceedings of the International Conference on Computer Vision Theory and Applications, Rome, Italy, 27–29 February 2016; Volume 5, pp. 143–149.
11. Wang, C.H.; Kuo, W.; Bensmail, H. Detection and classification of defect patterns on semiconductor wafers. *IIE Trans.* **2006**, *38*, 1059–1068. [CrossRef]
12. Luo, Q.; Fang, X.; Liu, L.; Yang, C.; Sun, Y. Automated Visual Defect Detection for Flat Steel Surface: A Survey. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 626–644. [CrossRef]
13. Li, C.; Li, J.; Li, Y.; He, L.; Fu, X. Fabric Defect Detection in Textile Manufacturing: A Survey of the State of the Art. *Secur. Commun. Netw.* **2021**, *2021*, 9948808. [CrossRef]
14. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556.
15. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1800–1807.
16. Westphal, E.; Seitz, H. A machine learning method for defect detection and visualization in selective laser sintering based on convolutional neural networks. *Addit. Manuf.* **2021**, *41*, 101965. [CrossRef]
17. Tabernik, D.; Sela, S.; Skvarc, J.; Skočaj, D. Deep-Learning-Based Computer Vision System for Surface-Defect Detection. In Proceedings of the International Conference on Virtual Storytelling, Thessaloniki, Greece, 23–25 September 2019.

18. Han, Y.; Yu, H. Fabric Defect Detection System Using Stacked Convolutional Denoising Auto-Encoders Trained with Synthetic Defect Data. *Appl. Sci.* **2020**, *10*, 2511. [CrossRef]
19. Peng, Y.; Ruan, S.; Cao, G.; Huang, S.; Kwok, N.; Zhou, S. Automated Product Boundary Defect Detection Based on Image Moment Feature Anomaly. *IEEE Access* **2019**, *7*, 52731–52742. [CrossRef]
20. Minhas, M.S.; Zelek, J.S. Anomaly Detection in Images. *arXiv* **2019**, arXiv:1905.13147.
21. Wang, L.; Zhang, D.; Guo, J.; Han, Y. Image Anomaly Detection Using Normal Data Only by Latent Space Resampling. *Appl. Sci.* **2020**, *10*, 8660. [CrossRef]
22. Lin, D.; Li, Y.; Xie, S.; Nwe, T.L.; Dong, S. DDR-ID: Dual deep reconstruction networks-based image decomposition for anomaly detection. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *14*, 2125–2139. [CrossRef]
23. Dai, W.; Erdt, M.; Sourin, A. Detection and segmentation of image anomalies based on unsupervised defect reparation. *Vis. Comput.* **2021**, *37*, 3093–3102. [CrossRef]
24. Beggel, L.; Pfeiffer, M.; Bischl, B. Robust Anomaly Detection in Images using Adversarial Autoencoders. *arXiv* **2019**, arXiv:1901.06355.
25. Zhu, Z.; Han, G.; Jia, G.; Shu, L. Modified DenseNet for Automatic Fabric Defect Detection With Edge Computing for Minimizing Latency. *IEEE Internet Things J.* **2020**, *7*, 9623–9636. [CrossRef]
26. Dafu, Y. Classification of Fabric Defects Based on Deep Adaptive Transfer Learning. In Proceedings of the 2019 Chinese Automation Congress (CAC), Hangzhou, China, 22–24 November 2019; pp. 5730–5733.
27. Pistori, H.; Amorim, W.P.; Martins, P.S.; Pereira, M.C.; Pereira, M.M.; Jacinto, M.A. Defect detection in raw hide and wet blue leather. In *Computational Modeling of Objects Represented in Images*; CRC Press: Boca Raton, FL, USA, 2006.
28. Moganam, P.K.; Sathia Seelan, D.A. Deep learning and machine learning neural network approaches for multi class leather texture defect classification and segmentation. *J. Leather Sci. Eng.* **2022**, *4*, 7. [CrossRef]
29. Moganam, P.K.; Sathia Seelan, D.A. Leather Defect Detection and Classification. 2022. Available online: https://www.kaggle.com/datasets/praveen2084/leather-defect-classification (accessed on 12 August 2022).
30. Moganam, P.K.; Seelan, D.A.S. Perceptron neural network-based machine learning approaches for leather defect detection and classification. *Instrum. Mes. Métrologie* **2020**, *19*, 421–429. [CrossRef]
31. Gan, Y.S.; Liong, S.; Wang, S.; Cheng, C.T. An improved automatic defect identification system on natural leather via generative adversarial network. *Int. J. Comput. Integr. Manuf.* **2022**, *35*, 1378–1394. [CrossRef]
32. Chen, S.; Cheng, Y.; Yang, W.; Wang, M. Surface Defect Detection of Wet-Blue Leather Using Hyperspectral Imaging. *IEEE Access* **2021**, *9*, 127685–127702. [CrossRef]
33. Parmar, N.; Vaswani, A.; Uszkoreit, J.; Kaiser, L.; Shazeer, N.M.; Ku, A.; Tran, D. Image Transformer. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018.
34. Jiang, Z.; Hou, Q.; Yuan, L.; Zhou, D.; Shi, Y.; Jin, X.; Wang, A.; Feng, J. All Tokens Matter: Token Labeling for Training Better Vision Transformers. *Neural Inf. Process. Syst.* **2021**, *34*, 18590–18602.
35. Wang, Y.; Huang, R.; Song, S.; Huang, Z.; Huang, G. Not All Images are Worth 16x16 Words: Dynamic Vision Transformers with Adaptive Sequence Length. *arXiv* **2021**, arXiv:2105.15075.
36. Mishra, P.; Verk, R.; Fornasier, D.; Piciarelli, C.; Foresti, G.L. VT-ADL: A Vision Transformer Network for Image Anomaly Detection and Localization. In Proceedings of the 2021 IEEE 30th International Symposium on Industrial Electronics (ISIE), Kyoto, Japan, 20–23 June 2021; pp. 1–6.
37. Xu, Y.; Zhang, Q.; Zhang, J.; Tao, D. ViTAE: Vision Transformer Advanced by Exploring Intrinsic Inductive Bias. *arXiv* **2021**, arXiv:2106.03348.
38. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 9992–10002.
39. Fang, Y.; Yang, S.; Wang, S.; Ge, Y.; Shan, Y.; Wang, X. Unleashing Vanilla Vision Transformer with Masked Image Modeling for Object Detection. *arXiv* **2022**, arXiv:2204.02964.
40. Deng, J.; Dong, W.; Socher, R.; Li, L.; Li, K.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
41. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
42. Wang, S.; Xia, X.; Ye, L.; Yang, B. Automatic Detection and Classification of Steel Surface Defect Using Deep Convolutional Neural Networks. *Metals* **2021**, *11*, 388. [CrossRef]
43. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
44. Google Cloud Developers. Advanced Guide to Inception V3. 2023. Available online: https://cloud.google.com/tpu/docs/inception-v3-advanced (accessed on 30 January 2023).
45. Tan, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv* **2019**, arXiv:1905.11946.

46. Bergmann, p.; Fauser, M.; Sattlegger, D.; Steger, C. A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 9584–9592. [CrossRef]
47. Honda, H. Vision Transformer Pipeline (Image). 2022. Available online: https://github.com/hirotomusiker/schwert_colab_data_storage/blob/master/images/vit_demo/vit_input.png (accessed on 23 March 2023).
48. Popescu, M.; Balas, V.E.; Perescu-Popescu, L.; Mastorakis, N.E. Multilayer perceptron and neural networks. *WSEAS Trans. Circuits Syst. Arch.* **2009**, *8*, 579–588.
49. Grosse, R. Lecture 5: Multilayer Perceptrons. inf. téc. In *Lecture Notes for the Course CSC321, "Intro to Neural Networks," for Undergraduates at the University of Toronto*; University of Toronto: Toronto, ON, Canada, 2019.
50. Lee, Y.; Kang, P. AnoViT: Unsupervised Anomaly Detection and Localization with Vision Transformer-based Encoder-Decoder. *IEEE Access* **2022**, *10*, 46717–46724. [CrossRef]