Computer Methods and Programs in Biomedicine 219 (2022) 106775

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.elsevier.com/locate/cmpb

Diffeomorphic transforms for data augmentation of highly variable shape and texture objects

Noelia Vallez[a,*], Gloria Bueno[a], Oscar Deniz[a], Saul Blanco[b]

[a] VISILAB, University of Castilla-La Mancha, E.T.S. Ingenieria Industrial, Avda. Camilo Jose Cela s/n, Ciudad Real 13071, Spain
[b] Institute of the Environment, University of Leon, Leon E-24071, Spain

ARTICLE INFO

Article history:
Received 27 September 2021
Revised 28 February 2022
Accepted 23 March 2022

Keywords:
Data augmentation
Diffeomorphism transforms
Algae classification
Taxon life cycle
Pollen classification
Glomeruli classification

ABSTRACT

Background and objective: Training a deep convolutional neural network (CNN) for automatic image classification requires a large database with images of labeled samples. However, in some applications such as biology and medicine only a few experts can correctly categorize each sample. Experts are able to identify small changes in shape and texture which go unnoticed by untrained people, as well as distinguish between objects in the same class that present drastically different shapes and textures. This means that currently available databases are too small and not suitable to train deep learning models from scratch. To deal with this problem, data augmentation techniques are commonly used to increase the dataset size. However, typical data augmentation methods introduce artifacts or apply distortions to the original image, which instead of creating new realistic samples, obtain basic spatial variations of the original ones.

Methods: We propose a novel data augmentation procedure which generates new realistic samples, by combining two samples that belong to the same class. Although the idea behind the method described in this paper is to mimic the variations that diatoms experience in different stages of their life cycle, it has also been demonstrated in glomeruli and pollen identification problems. This new data augmentation procedure is based on morphing and image registration methods that perform diffeomorphic transformations.

Results: The proposed technique achieves an increase in accuracy over existing techniques of 0.47%, 1.47%, and 0.23% for diatom, glomeruli and pollen problems respectively.

Conclusions: For the Diatom dataset, the method is able to simulate the shape changes in different diatom life cycle stages, and thus, images generated resemble newly acquired samples with intermediate shapes. In fact, the other methods compared obtained worse results than those which were not using data augmentation. For the Glomeruli dataset, the method is able to add new samples with different shapes and degrees of sclerosis (through different textures). This is the case where our proposed DA method is more beneficial, when objects highly differ in both shape and texture. Finally, for the Pollen dataset, since there are only small variations between samples in a few classes and this dataset has other features such as noise which are likely to benefit other existing DA techniques, the method still shows an improvement of the results.

1. Introduction

Training the necessary models to automatically classify samples requires large databases with images of already labeled samples. This is not a serious problem in some applications such as vehicle detection, face detection or scene recognition where there are large datasets already available and people can easily label more samples if necessary. However, in other applications, collecting these datasets is difficult and requires an expert to correctly categorize each sample. This is especially challenging in biology or pathology where only trained and experienced specialists can label the data. Furthermore, in these cases, currently available databases can not solve the problem since they are too small to train deep learning models.

One of these challenging applications is diatom taxa identification. Diatoms are a group of microalgae (a type of phytoplankton)

* Corresponding author.
E-mail address: noelia.vallez@uclm.es (N. Vallez).

https://doi.org/10.1016/j.cmpb.2022.106775

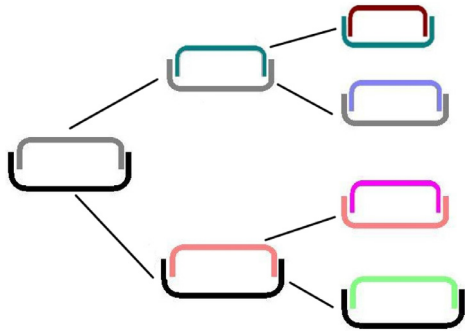


Fig. 1. Diatom asexual reproduction. Cell size is reduced in successive generations except for the cell that conserves the original epitheca. Best viewed in color.

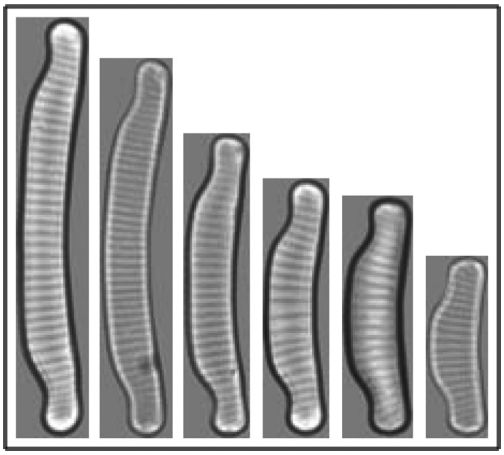


Fig. 2. Size reduction in the life cycle of the *Eunotia tenella*.

that are part of the aquatic ecosystem and thus, can be found in aquatic environments such as oceans and rivers.

In diatom identification, expert biologists take into account the differences in the morphometric characteristics (length, width, etc.) and the frustule striae of the microalgae. However, these shape and texture features change during the so-called life cycle of the diatom. Thus, the life cycle plays an important role in diatom identification and biologists focus on identifying these changes [1].

Variations in the diatom life cycle are caused by its reproduction. The reproduction process can occur by either sexual or asexual methods and differs according to the diatom shape although the primary form of reproduction in diatoms is asexual [2]. It occurs by binary fission where the cell is divided in half, separating its two valves. Later on, new siliceous valves are deposited on the naked sides. Both valves, called *epitheca* and *hypotheca*, of the mother cell behave as the epitheca in the daughter cells. Thus, new diatoms where the hypotheca behaves as epitheca are smaller while the others retain the parental size (see Fig. 1). As a result, the average size of the diatom population decreases with successive divisions. This phenomenon is known as the MacDonald–Pfitzer hypothesis. Fig. 2 shows the life cycle of a diatom.

As a result, the changes in texture and shape caused by the life cycle, the large number of diatoms per water sample, and the vast number of species, make manual identification of these organisms a tedious task [3]. Moreover, some of the species are easily confused (see Section 2.1.1).

The existence of thousands of species makes deep learning techniques suitable to automatically perform the identification. However, as explained before, the number of labeled images per class is not enough [4] and the life cycle information should also be used to train the models [5].

To alleviate the limitations of small datasets, Data Augmentation can help reduce the class imbalance problem by oversampling the training dataset. Over the last few years, several Data Augmentation approaches have been proposed [6]. A very basic Data Augmentation technique is to apply simple image manipulations to the input images such as: geometric transformations (flipping, rotations, translations, crops...) [7] and simple image processing (color space changes, noise injection...). Geometric transformations are the easiest to implement and are also the most used since they are included in popular deep learning frameworks such as Keras or PyTorch. They provide satisfactory results when positional biases are present in the training data. The only aspect to be concerned about is the fact that they do not always preserve the class labels after the transformation is applied. On the contrary, this is not a problem when applying image processing modifications. For example, noise injection can help the network learn more robust features [8].

Kernel filters have also been used to sharpen and blur training images [9]. Intuitively, blurring adds robustness to motion blur and non focused images. On the other hand, sharpening adds the possibility of encapsulating small details of the objects. However, kernel filters can be added to the network architecture instead of being applied directly to the dataset. In that case, the training will optimize their weights along with the rest of the network weights.

Random erasing [10] is another Data Augmentation technique inspired by the mechanisms of dropout regularization which, instead of using the output of a layer as input data, this output is directly applied over the training dataset. Of course, this method also presents issues when the part of the image that contains the key discriminant information is erased.

Mixing images together has also been used for Data Augmentation. Mixing images can be as simple as averaging the pixel values of a pair of images or inserting part of an image into the other one, or as complex as training and using a Generative Adversarial Network (GAN) to generate mixed images. This is not as intuitive as the above approaches, but has been demonstrated to increase the performance of CNNs [11]. However, for specialist domains the resulting images may make no sense.

In addition to mixing images, GAN-based methods have also been proposed to create artificial instances from a dataset following the sample distribution [12]. Although this method is very promising, if a GAN is trained on a given dataset, it will learn the information represented in that dataset and will generate data in the same space. Thus, a GAN is not adding any new information to the dataset although increasing the dataset size helps CNNs to generalize. Moreover, in order to match the target domain distribution, GANs can artificially create image features which can lead to misdiagnosis in medical applications [13].

Adversarial training can be applied to improve the CNN performance [14]. In this case, the dataset is not updated with new samples and the artificial images are used to improve robustness to adversarial attacks.

Other authors have proposed to learn spatial transformation from a large class of diffeomorphisms [15]. The problem of this approach is that it is class-dependendent and requires to model each class in the dataset. Moreover, it was only demonstrated on MNIST and using a very small CNN.

Following the same line, diffeomorphic image transformations have also been applied for data augmentation in MRI segmentation [16]. In that case, the authors proposed to obtain a mean template from the samples and then use the sampled transformations to alter training data. To achieve that, they use a Hamiltonian Monte Carlo (HMC) scheme.

Wavelet and constant-Q Gabor transforms have also been applied to perform data augmentation improving geometric and image perturbation aprroaches [17].

Finally, morphing has often been applied when dealing with facial recognition problems, more specifically in morphing attacks, to improve face recognition models but has not been applied to other datasets [14].

In this work, to cope with the problem of having small datasets, we use *transfer learning* and data augmentation techniques. Both have been demonstrated to be effective in order to reduce overfitting and improve generalization [6]. The goal of our study is to perform a data augmentation step that generates new realistic samples to help CNNs generalize and improve their results. For this purpose, we propose a procedure, inspired by the diatom life cycle, that simulates samples in different stages by means of diffeomorphic transforms. The proposed data augmentation method is then based on:

- Image morphing
- Image registration
  - Stationary Velocity Field
  - Diffeomorphic Log-Demons
  - B-Spline Composition and Level Sets
  - Matching CNNs

To the best of our knowledge, this is the first time these methods are applied for data augmentation.

In order to demonstrate the applicability of the proposed data augmentation technique, three small datasets from different tasks have been used: Glomerulosclerosis identification, diatom classification and pollen classification. A comparison between the results obtained with 6 CNN architectures with three existing data augmentation methods (geometric transformations, noise injection and artificial image generation with GANs) and our approach has been carried out.

2. Materials and methods

2.1. Datasets

Three datasets from different domains have been employed in this work. The first one is composed of different diatom taxa. Although there are other diatom datasets which are larger, to the best of our knowledge only the one used in this work ensures that different stages of the life cycle are represented.

The second dataset contains images from several classes of pollen. Although pollen species do not have a life cycle like diatoms, there is also high variability between samples.

Finally, the third dataset contains normal and sclerosed glomeruli samples employed in nephropathology studies [18]. Like pollen, glomeruli do not have a life cycle but different cuts of this anatomic structure can be seen as different stages of a life cycle. Moreover, texture also plays an important role for identification in this case.

2.1.1. Diatom dataset

The database used in this work is a collection of 976 diatom images from 14 different species with between 40 and 121 images[1] per class. Images from each class represent diatoms at different stages of their life cycle.

Samples from *Gomphonema minutum*, *Luticola goeppertiana*, *Nitzschia amphibia*, and *Nitzschia capitellata* were obtained from the AQUALITAS project [19]. Images were captured with a Brunel SP30 monocular microscope with standard Brunel DIN parfocal objectives of 60× (0.85 NA) and 100× (1.25 NA) using a LED with white light ($\lambda = 442$ nm). A Brunel Digicam LCMOS 5 Mpixel camera was used for image acquisition. The image resolution was 2,592 × 1,944 pixels.

Table 1
Number of images per class in the diatom dataset.

Class	#Total	#Train	#Val.	#Test
001 - Eunotia tenella	68	54	7	7
002 - Fragilariforma bicapitata	90	72	8	10
003 - Gomphonema augur	90	71	10	9
004 - Stauroneis smithii	75	59	9	7
005 - Gomphonema minutum	69	55	7	7
006 - Luticola goeppertiana	74	61	6	7
007 - Nitzschia capitellata	79	64	6	9
008 - Nitzschia amphibia	49	38	6	5
009 - Sellaphora pupula	40	32	4	4
010 - Sellaphora obesa	72	58	7	7
011 - Sellaphora blackfordensis	56	44	6	6
012 - Sellaphora capitata	121	97	12	12
013 - Sellaphora auldreekie	40	32	4	4
014 - Sellaphora lanceolata	53	43	5	5
Total	976	780	97	99

Table 2
Number of images in the polen dataset.

#Total	#Train	#Val.	#Test
2591	1701	445	445

Eunotia tenella, *Fragilariforma bicapitata*, *Gomphonema augur*, and *Stauroneis smithii* were obtained from the DIADIST dataset [20].

The rest of the samples were captured using a digital camera, Kodak MegaPlus ES1.O, with 1008 × 1018 pixels resolution [21]. The camera was attached to an Axiophot photomicroscope (Zeiss) with a 100× apochromatic oil immersion lens (1.4 NA) and a ×1.6 Optivar magnification changer. Bright field optics were used throughout the whole process.

To validate and test the models obtained, the dataset has been randomly divided into 3 sets: training, validation, and test. Both validation and test sets contain around 10% of the samples whereas the remaining 80% is used for training. Table 1 summarizes how samples are distributed across all classes and sets and Fig. 3 shows an example of each class.

2.1.2. Pollen dataset

The identification of pollen grains is useful in honey quality control, crime scene identification, and the study of the paleoenvironment through fossils [22,23].

The pollen classification dataset employed in this work is POLLEN73S [24] plus two additional classes from POLEN23E, syagrus and arecaceae [22].

POLLEN73S is composed of 2523 color images from 73 different categories with an average resolution of 512 × 512. All pollen samples belong to the Campo Grande City urban area in the Brazilian Savannah. The microscope used to capture the images is a Carl Zeiss Microimaging microscope equipped with 40× objective lenses. Images have been acquired at different angles. Each of the 73 pollen types contains 35 samples except gomphrena sp which has 10, trema micrantha which has 34, and zea mays which has 29.

On the other hand, the two additional classes were collected from the same area and under the same conditions. They also contain 35 samples per category. That leads to a total of 2593 samples from 75 pollen types (Table 2).

Fig. 4 shows an example of each type. Similarly to the diatom dataset, training, validation and test sets contain 80%, 10%, and 10% of the samples respectively.[2]

[1] Images available in https://doi.org/10.6084/m9.figshare.18551300.v2.

[2] Images available in https://doi.org/10.6084/m9.figshare.18587321.v1.

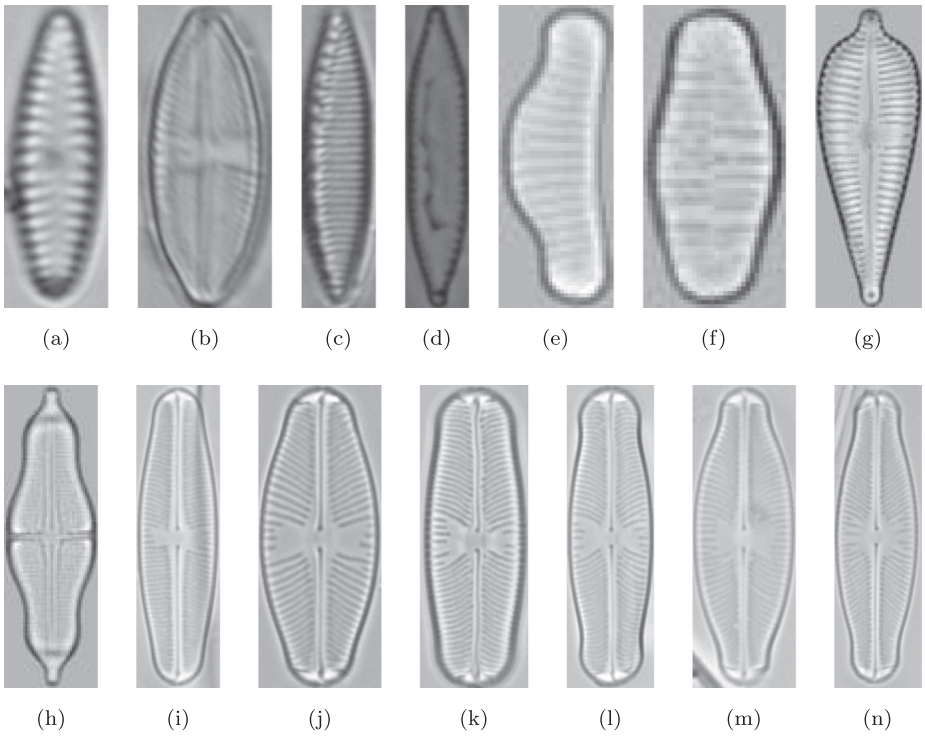


Fig. 3. Image samples from the dataset. a) *Gomphonema minutum*, b) *Luticola goeppertiana*, c) *Nitzschia amphibia*, d) *Nitzschia capitellata*, e) *Eunotia tenella*, f) *Fragilariforma bicapitata*, g) *Gomphonema augur*, h) *Stauroneis smithii*, i) *Sellaphora pupula*, j) *Sellaphora obesa*, k) *Sellaphora blackfordensis*, l) *Sellaphora capitata*, m) *Sellaphora auldreekie*, n) *Sellaphora lanceolata*.

Table 3
Number of images per class in the glomeruli dataset.

Class	#Total	#Train	#Val.	#Test
Sclerosed	502	350	76	76
Normal	824	576	124	124
Total	1326	926	200	200

2.1.3. Glomeruli dataset

Glomeruli are clusters of capillaries responsible for eliminating unnecessary substances from the human body. Glomerular lesions present the so-called glomerulosclerosis, which is characterized by glomeruli with different degrees of sclerosis depending on how much of their area is affected [18].

The dataset employed here is part of the AIDPATH kidney database acquired and digitized from three European institutions: Castilla-La Mancha's Healthcare services (Spain), Andalusian Health Service (Spain) and Vilnius University Hospital Santaros Klinikos (Lithuania) [18]. Tissue samples were collected with a 100 μm–300 μm biopsy needle. Paraffin blocks were then prepared using tissue sections of 4 μm and stained using Periodic Acid Schiff (PAS). Digital whole slide image (WSI) acquisition was performed with the Leica Aperio ScanScope CS scanner at 20× magnification. As a result, a dataset of 47 kidney WSIs was obtained[3].

From the digital WSIs a set of 1326 annotated glomeruli was obtained including 502 sclerosed or semi-sclerosed samples and 824 normal samples. The average image resolution was 250 × 250 pixels. Fig. 5 shows an example of each class and Table 3 shows how the dataset is distributed across train, validation and test sets.

[3] Images available in https://doi.org/10.6084/m9.figshare.18586565.v1.

2.2. Data augmentation

The so-called *data augmentation* technique deals with the most frequently reported problems of deep neural networks training: the lack of a sufficient amount of training images. This is also aggravated when the datasets are later reduced by partitioning them into training, validation, and test sets.

To deal with this problem, four different methods of data augmentation have been tested and compared. The first one is based on the use of geometric transformations such as flips or rotations. This approach has been widely employed in the literature and can be used in general for any type of dataset. The second method is based on noise injection. In some cases, noise can leverage the model's generalization ability. The third method uses a GAN to generate new artificial samples. Finally, a novel data augmentation technique that performs diffeomorphic transformations and generates a set of intermediate images from every pair of samples through morphing and registration techniques is used.

2.2.1. Geometric transformations

Among all the data augmentation techniques, applying geometric transformations is the easiest way of enlarging a dataset. However, not all transformations are suitable to be used for this purpose. Generated images should simulate real images taken under a limited set of possible conditions. Thus, the following operations have been selected to carry out the data augmentation step:

1. Horizontal flip
2. Vertical flip
3. Rotations between 0° and 90°

The combination of these three transformations is randomly applied each time a batch is requested during training. The validation and test sets remain unaltered. After this process, images are resized to the network input size. Fig. 6 shows some examples

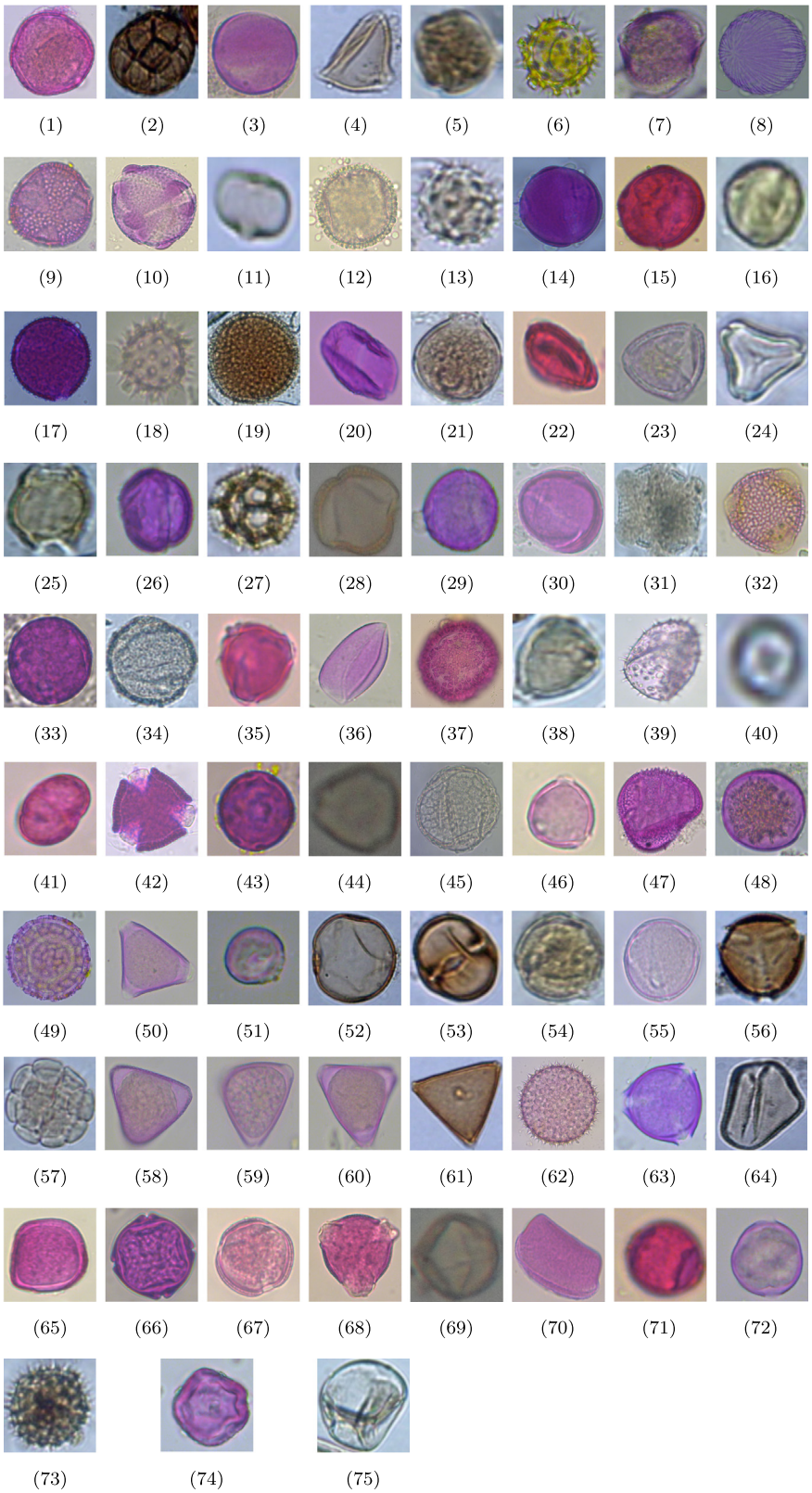
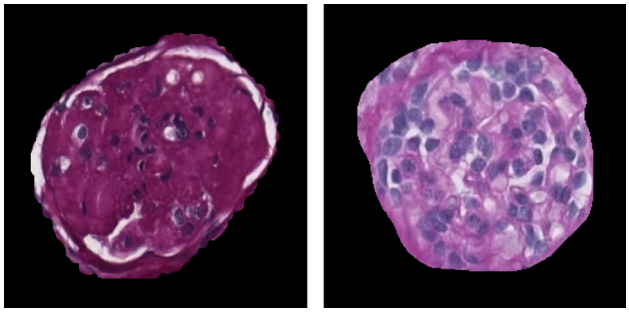


Fig. 4. Image samples from the pollen dataset. 1) *Acrocomia aculeta*, 2) *Anadenanthera colubrina*, 3) *Arachis sp*, 4) *Arecaceae*, 5) *Arrabidaea florida*, 6) *Aspilia gracielae*, 7) *Bacopa australis*, 8) *Beladona*, 9) *Caesalpinia peltophoroides*, 10) *Caryocar brasiliensis*, 11) *Cecropia pachystachya*, 12) *Ceiba speciosa*, 13) *Chromolaena laevigata*, 14) *Cissus campestris*, 15) *Cissus spinosa*, 16) *Combretum discolou*, 17) *Cordia trichotoma*, 18) *Cosmos caudatus*, 19) *Croton urucurana*, 20) *Dianella tasmanica*, 21) *Dipteryx alata*, 22) *Doliocarpus dentatus*, 23) *Erythrina mulungu*, 24) *Eucalyptus sp*, 25) *Faramea sp*, 26) *Genipa auniricana*, 27) *Gomphrena sp*, 28) *Guapuruvu*, 29) *Guazuma ulmifolia*, 30) *Hortia oreadica*, 31) *Hyptis sp*, 32) *Ligustrum lucidum*, 33) *Luehea divaricata*, 34) *Mabea fistulifera*, 35) *Machaerium aculeatum*, 36) *Magnolia champaca*, 37) *Manihot esculenta*, 38) *Matayba guianensis*, 39) *Mauritia flexuosa*, 40) *Mimosa ditans*, 41) *Mimosa pigra*, 42) *Mitostemma brevifilis*, 43) *Myracroduon urundeuva*, 44) *Myrcia guianensis*, 45) *Ochroma pyramidale*, 46) *Ouratea hexasperma*, 47) *Pachia aquatica*, 48) *Palmeira real*, 49) *Passiflora giberti*, 50) *Paullinia spicata*, 51) *Piper aduncum*, 52) *Poaceae sp*, 53) *Protium heptaphyllum*, 54) *Qualea multiflora*, 55) *Ricinus communis*, 56) *Schinus sp*, 57) *Senegalia plumosa*, 58) *Serjania erecta*, 59) *Serjania hebecarpa*, 60) *Serjania laruotteana*, 61) *Serjania sp*, 62) *Sida cerradoensis*, 63) *Solanum sisymbrifolium*, 64) *Syagrus*, 65) *Syagrus romanzoffiana*, 66) *Symplocos nitens*, 67) *Tabebuia chysotricha*, 68) *Tabebuia rosealba*, 69) *Tapirira guianensis*, 70) *Tradescantia Pallida*, 71) *Trema micrantha*, 72) *Trembleya phlogiformis*, 73) *Tridax procumbens*, 74) *Vochysia divergens*, 75) *Zea mays*.. Best viewed in color.



(a) Sclerosed Glomerulus (b) Normal Glomerulus

Fig. 5. Image samples from the glomeruli dataset.

of where these transformations are applied to the three datasets used.

2.2.2. Noise injection

The noise injection method refers to artificially adding noise to the CNN input data during training. This causes the training data to jitter in the feature space during training, making it harder for the CNN to find a solution that exactly matches the original training dataset, reducing overfitting and improving generalization. The noise vector usually comes from a probability density function. In this case, we have added noise following a Gaussian distribution with a mean of 0.1 and a standard deviation of 0.8. Fig. 7 shows examples of the noise added to the three datasets considered.

2.2.3. GAN-based image generation

The architecture used in the Generative Adversarial Network (GAN) is formed by a generator and a discriminator. The former is responsible for producing images as close as possible to those contained in the training dataset. The latter takes both training set and generator images, trying to score and discern whether they are real or not. In the training process, both networks are optimized at the same time. As a result, there is a competitive training scheme, in which the generator tries to trick the discriminator producing better real-like images, while the discriminator is continuously improving its ability to detect those images. After training a GAN architecture with a dataset, it is possible to use the generator to artificially generate new samples.

In this work, we have used a generator composed of 5 blocks. Each one contains a batch normalization, a rectified linear unit activation and a transposed convolution with 512, 256, 128, 64 and 3 filters (the last one produces the image, and therefore the 3 filters correspond to the color channels). At the top of the generator, the latent vector (the features used as input for the network) is initialized randomly, with a size equal to the input ($128 \times 128 \times 3 = 46,152$ features). On the other side, the discriminator reproduces the same architecture, but inverted. For this purpose, each block contains regular convolution layers and the output is a single neuron to predict whether the input is real or fake.

Fig. 8 shows images generated with each of the 3 GAN models obtained.

2.2.4. Proposed methods

We propose to add new samples that simulate intermediate phases of a diatom's life cycle. To achieve this goal, a data augmentation approach based on image morphing and image registration has been followed.[4]

Image morphing and morphable 3D models have been frequently applied to generate synthetic face images for visual effects or face recognition problems [25]. A sequence is generated to obtain a transition between the two images.

On the other hand, the aim of image registration is to find a spatial transformation T such that:

$$I_1 \approx I_0 \circ T \tag{1}$$

where I_1 denotes the target image and I_0, the source image.

From each pair of samples, it is possible to generate samples in an intermediate state of the life cycle using both techniques.

1. Morphing
   Given a pair of images I and J, the first step of morphing establishes a pixel correspondence between each pixel (x_i, y_i) in I and (x_j, y_j) in J. Then, each pixel of the morphed image M, (x_m, y_m), is given by:

$$\begin{aligned} x_m &= (1-\alpha)x_i + \alpha x_j \\ y_m &= (1-\alpha)y_i + \alpha y_j \end{aligned} \tag{2}$$

   and the pixel intensity is obtained as:

$$M(x_m, y_m) = (1-\alpha)I(x_i, y_i) + \alpha J(x_j, y_j) \tag{3}$$

[4] The code can be found in https://github.com/noeliavallez/DiffeomorphicDA.

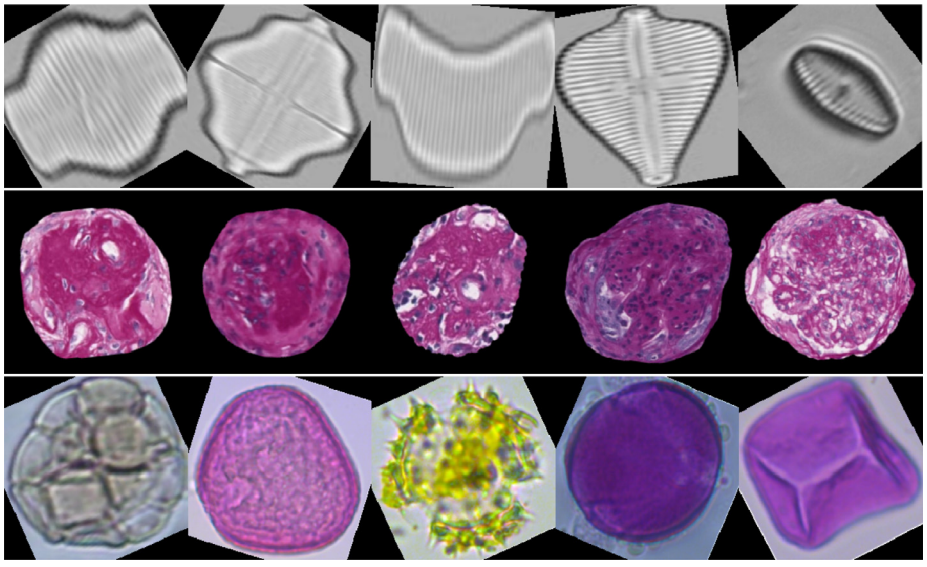


Fig. 6. General data augmentation examples.

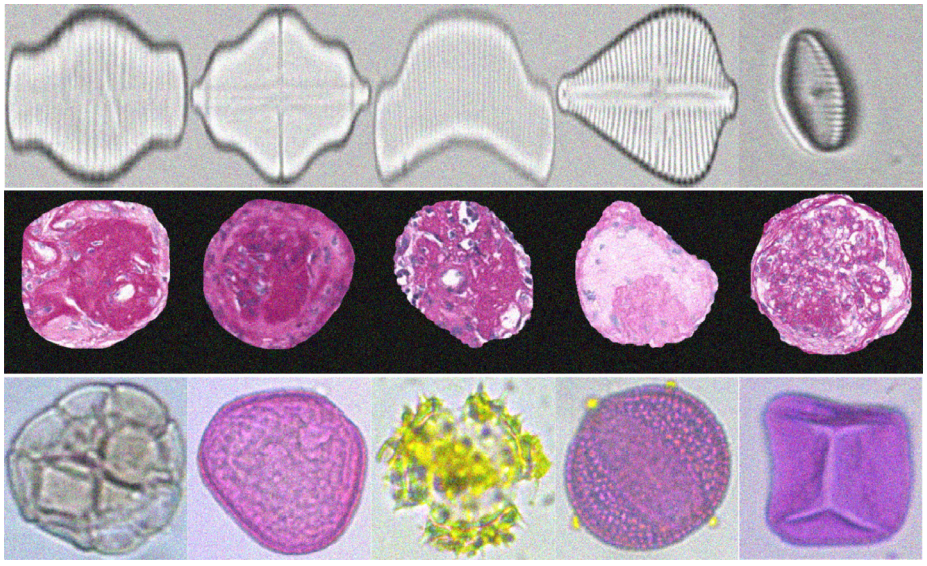


Fig. 7. Noise injection data augmentation examples.

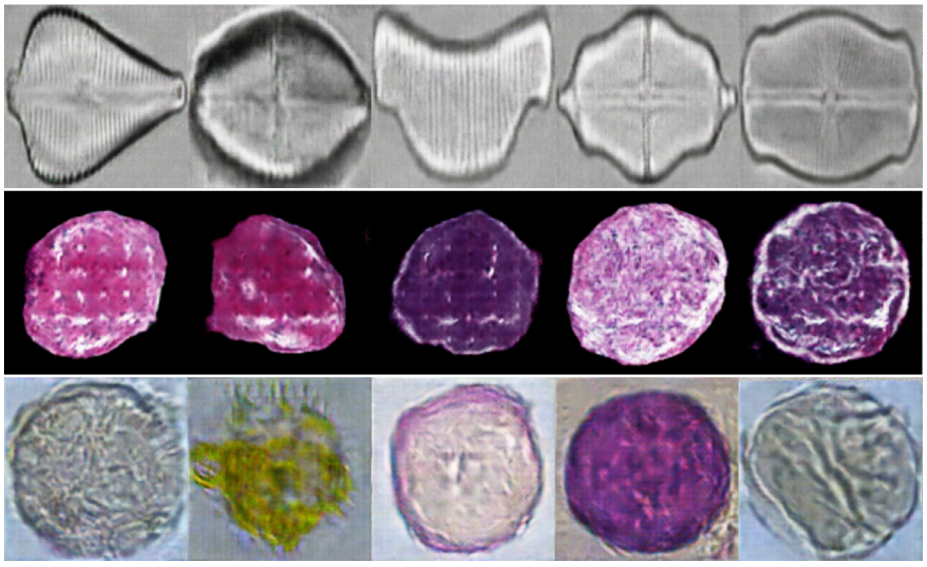


Fig. 8. GAN-based data augmentation examples.

where $\alpha \in [0, 1]$ is a parameter that controls how close M is to images I and J. When α is 0, $M = I$ and if α is 1 $M = J$.
To find the correspondence between all pixels, a set of keypoints needs to be selected first. These points are pixels from I and J which are known to be correspondents. For example, in face morphing the center of the eyes are two keypoints that are usually selected in both images. In this case a total of 36 keypoints are established around the diatom contour and the image border. These keypoints are:

- The corners of the image
- Three points equally distributed between each adjacent corner pair
- From the center of the image, the leftmost, rightmost, uppermost, bottommost points of the diatom contour
- Four points between each of the previous adjacent pairs of the diatom contour

The number of keypoints used have been selected empirically. Using less points does not obtain realistic enough images and using more points only increases the complexity of the method.

To extract the diatom contour, Elliptical Fourier descriptors (EFD) were used. These descriptors have demonstrated their suitability to describe the diatom contour [5]. The method employed was presented in Kuhl and Giardina [26] and obtains the Fourier coefficients of a chain-encoded contour. Therefore, the first step in the algorithm is to obtain an initial contour image where the Freeman chain code is computed. This is achieved with image thresholding in this case. Once the initial chain-encoded contour is obtained, the changes in x and y projections, Δx_i and Δy_i, can be obtained as follows:

$$\Delta x_i = sgn(6 - a_i)sgn(2 - a_i) \tag{4}$$

$$\Delta y_i = sgn(4 - a_i)sgn(a_i) \tag{5}$$

$$\Delta t_i = 1 + \left(\frac{\sqrt{2} - 1}{2}\right)(1 - (-1)^{a_i}) \tag{6}$$

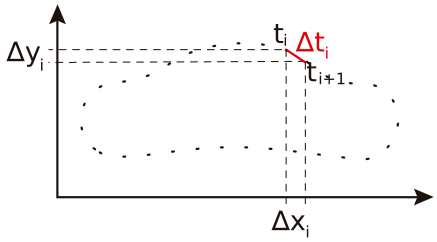


Fig. 9. Freeman chain code projections.

Fig. 10. Keypoints and Delaunay triangulation.

where a_i is the ith element in the Freeman chain code obtained and Δt_i is the modulus of the segment between two adjacent points (Fig. 9).

Then, $x(t)$ and $y(t)$ functions (for the x and y projections respectively) can be approximated as two Fourier series, and their corresponding Fourier coefficients, a_n, b_n, c_n, and d_n, can be obtained through the following equations:

$$a_n = \frac{T}{2n^2\pi^2}\sum_{p=1}^{K}\frac{\Delta x_p}{\Delta t_p}\left[cos\frac{2n\pi t_p}{T} - cos\frac{2n\pi t_{p-1}}{T}\right] \tag{7}$$

$$b_n = \frac{T}{2n^2\pi^2}\sum_{p=1}^{K}\frac{\Delta x_p}{\Delta t_p}\left[sin\frac{2n\pi t_p}{T} - sin\frac{2n\pi t_{p-1}}{T}\right] \tag{8}$$

$$c_n = \frac{T}{2n^2\pi^2}\sum_{p=1}^{K}\frac{\Delta y_p}{\Delta t_p}\left[cos\frac{2n\pi t_p}{T} - cos\frac{2n\pi t_{p-1}}{T}\right] \tag{9}$$

$$d_n = \frac{T}{2n^2\pi^2}\sum_{p=1}^{K}\frac{\Delta y_p}{\Delta t_p}\left[sin\frac{2n\pi t_p}{T} - sin\frac{2n\pi t_{p-1}}{T}\right] \tag{10}$$

where K is the number of harmonics used, n is the order of the harmonic coefficients and T is the perimeter.

Finally, the amplitude of the nth harmonic is calculated as:

$$amp_n = \frac{1}{2}\sqrt{a_n^2 + b_n^2 + c_n^2 + d_n^2} \tag{11}$$

Once the diatom contour is approximated, the aforementioned keypoints are selected in both images. Then, their corresponding position in the output image M is calculated as intermediate points as shown in Eq. (2). With all keypoints selected, the Delaunay triangulation is obtained for all 3 images (I, J and M) using the keypoints as vertices for all the triangles that can be formed (Fig. 10). The affine transforms between each triangle in M and the triangles in I and J that are formed by the same keypoints are computed.

Finally, the process iterates over all pixels of M, determining which triangle the pixel belongs to, applying the corresponding affine transforms to find the coordinates of the same pixel in I and J, and calculating the pixel color as shown in Eq. (3).

Fig. 11 shows an example of the image sequence generated from a pair of samples.

2. Stationary Velocity Field Algorithm

Stationary Velocity Field (SVF) is a non-parametric image registration technique used to estimate deformations between image pairs [27,28]. For a given source-target image pair in a registration problem (I_0, I_1), it obtains a spatial transformation, T, such that $I_0 \circ T \approx I_1$. The approach followed in this work is the one in Niethammer et al. [29].

Image registration algorithms are often expressed as an optimization problem:

$$\gamma^* = \underset{\gamma}{\operatorname{argmin}}\, \lambda Reg\left[\Phi^{-1}(\gamma)\right] + \mathrm{Sim}[I_0 \circ \Phi^{-1}(\gamma), I_1] \tag{12}$$

where Φ is the deformation, γ parametrize Φ, $\lambda > 0$, $Reg[\cdot]$ is a penalty that facilitates spatially regular deformations, and $Sim[\cdot,\cdot]$ is used to penalize differences between two images [27].

SVF is a fluid-type registration method where the deformation Φ is obtained via time-integration of a velocity field $v(x,t)$, which has to be approximated. The governing differential equation is: $\Phi_t(x,t) = v(\Phi(x,t),t)$, $\Phi(x,0) = \Phi_{(0)}(x)$, where $\Phi_{(0)}$ is the initial map.

To obtain a diffeomorphic transform, the non-smoothness of the velocity field, v, is penalized:

$$v^* = \underset{v}{\operatorname{argmin}}\, \lambda \int_0^1 \|v\|_L^2 dt + \mathrm{Sim}[I_0 \circ \Phi^{-1}(1), I_1],$$

$$\Phi_t^{-1} + J\Phi^{-1}v = 0 \quad and \quad \Phi^{-1}(0) = \mathrm{id} \tag{13}$$

where J is the Jacobian, λ is a constant > 0, Sim is the Normalized Cross Correlation similarity measure, and $\|v\|_L^2 = \langle L^\dagger Lv, v\rangle$ is a spatial norm defined by specifying the differential operator L and its adjoint $L^\dagger$.

Since the vector-valued momentum, m, is equal to $L\dagger Lv$, $\|v\|_L^2 = \langle m, v\rangle$. As a result, the authors proposed to formulate the problem to optimize it over the vector momentum, m_0:

$$m^* = \underset{m_0}{\operatorname{argmin}}\, \lambda\langle m_0, v_0\rangle + \mathrm{Sim}[I_0 \circ \Phi^{-1}(1), I_1],$$

$$\Phi_t^{-1} + D\Phi^{-1}v = 0, \quad \Phi^{-1}(0) = \mathrm{id} \quad and \quad v_0 = \left(L^\dagger L\right)^{-1} m_o \tag{14}$$

The method has been applied to each diatom pair in the 14 diatom species. Thus, the registration problem considers each possible pair of images from the same species as the source and target images.

Prior to applying the method to diatom images, the following steps were taken to prepare the data:

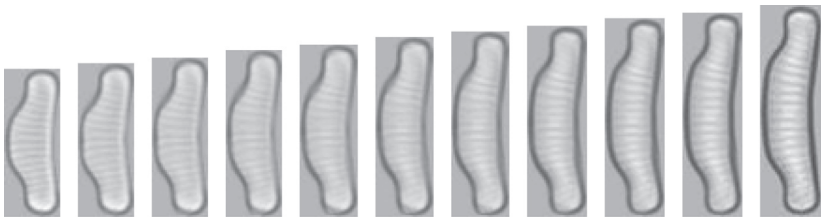
(a) A pair of images is selected from the same class
(b) The smallest image in the pair is then resized to the size of the largest one
(c) A padding of sixteen pixels is added in both images since it improves the resulting deformations

Fig. 12 shows some examples obtained with this method.

3. Diffeomorphic Log-Demons Registration

The concept of demons was first introduced by Maxwell in the 19th century to illustrate a paradox in thermodynamics. Consider a semipermeable membrane that separates a gas composed of two types of particles. This membrane will contain a set of "demons", which are able to distinguish between them and will diffuse one type of particle to one side of the membrane and the other type to the other side. This system contradicts the second principle of thermodynamics as it produces an entropy reduction. However, demons generate a large amount of entropy recognizing the particle type, thus solving the paradox.

In image registration, demons-based methods assume that the contour of an object inside an image is a membrane. Then, demons are scattered in the image contours. Demons registration utilizes optical flow equation as basis forces with the purpose of finding tiny deformations. For a point p in space, let



(a) OI1 (b) M1 (c) M2 (d) M3 (e) M4 (f) M5 (g) M6 (h) M7 (i) M8 (j) M9 (k) OI2

Fig. 11. Example of data augmentation using morphing. a) and k) are images from the original dataset and b)–j) is the image sequence generated between them.

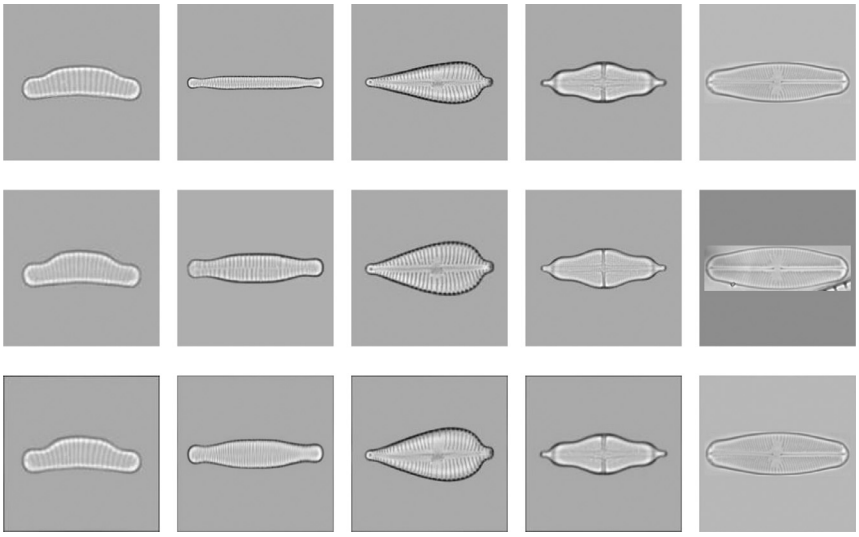


Fig. 12. Example of data augmentation using registration through SVF algorithm. The first row corresponds to the source images, the second row to the target images, and the last row to the warped images.

i_1 and i_0 be intensity values in target image I_1 and source image I_0 respectively. According to [30], Eq. (15) can be used to compute the velocity, u, that matches the point p to its corresponding point in $I1$. Δf represents the internal edge force and is calculated as the gradient image of the target image, $(i_0 - i_1)$ is the external force, and the term $(i_0 - i_1)^2$ is added to make the equation more stable and suitable for registration.

$$u = \frac{(i_0 - i_1)\Delta i_1}{|\Delta i_1|^2 + (i_0 - i_1)^2} \tag{15}$$

Diffeomorphic Log Demons use a diffeomorphic transformation ϕ related to the exponential map of the velocity field ν : $\phi = \exp(\nu)$.

The method used in this work implements the demons algorithm described in Thirion [30] that employs the classical operations from Demons algorithms, but the transformation applied to the source image is defined as an exponential velocity field. This concept is referred to as log-domain in Vercauteren et al. [31]. The general procedure consists of two steps. Whereas the first step looks for the unconstrained update for the velocity field, the second one applies a simple Gaussian smoothing filter on the update transformation recently computed.

The global energy equation used in this algorithm consists of two elements: a similarity criterion, E_{Sim} which is used to measure the likeness between the target image, I_1, and the source image, I_0, and a regularization energy component, E_{Reg}, given by the spatial transformation applied:

$$E = E_{Sim} + E_{\text{Reg}} = \frac{\sum (I_1 - I_0)^2}{area} + \frac{\sum Jac^2}{area} \tag{16}$$

where Jac corresponds to the Jacobian determinant of the exponential velocity field and $area$ is the size of the images in pixels.

Finally, this method is characterized by using a multi-resolution registration procedure. The method starts by registering the source image with the lowest resolution given as an input parameter. In this method, this input value, n, corresponds to the multi-resolution levels used. Then, the registration is applied again on the warped image obtained in the previous step, but using a higher resolution. The lowest possible image resolution is equal to $\frac{1}{2^n}$ times the initial one. Therefore, this method doubles the image resolution each time the registration algorithm is applied. Fig. 13 shows an example of the registration applied with 3 levels of multi-resolution.

The algorithm has been applied following these steps [32]:

(a) Scale both images to the corresponding resolution given by the multi-resolution registration level. Since this is an iterative process, the scaling applied for each image is computed as described above, beginning with the lowest resolution.
(b) Choose a starting spatial transformation, ϕ. The initial one corresponds to a neutral matrix transformation.
(c) Apply the spatial transformation defined in the previous step to the source image I_0 and update its value.
(d) Compute the normalized gradient, J, of the difference between target and source images, $diff$, and use them and the weights on the similarity term, σ_i, and spatial uncertainties, σ_x, to obtain the update (Eq. (17)). Then, the parameters to calculate the new velocity field are obtained as shown in

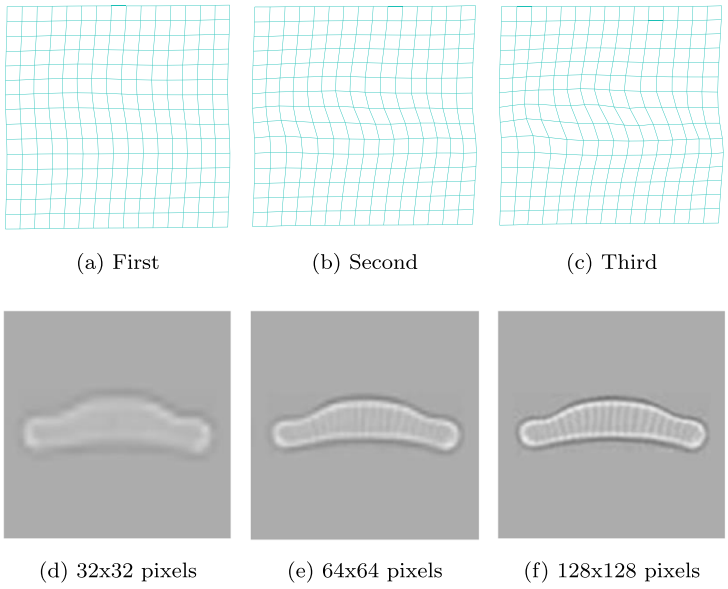


Fig. 13. Transformation matrices (a)–(c) and results (d)–(f) of a 3-level registration (each matrix is applied to their corresponding image below). The first column corresponds to the lowest level of resolution and the last column corresponds to the highest level.

Eq. (18) where g_x and g_y represent the gradient of I_0.

$$update = \frac{diff}{\|J\|^2 + diff^2 \frac{\sigma_i^2}{\sigma_x^2}} \tag{17}$$

$$u_x = g_x \cdot update \quad u_y = g_y \cdot update \tag{18}$$

(e) Perform the regularization of u_x and u_y parameters with a Gaussian smoothing filter (Eq. (19)), and compute the final spatial transformation (Eq. (20)).

$$v'_x = v_x + \sigma_x \cdot u_x \quad v'_y = v_y + \sigma_x \cdot u_y \tag{19}$$

$$\phi = exp(v) \tag{20}$$

Steps 3–5 are repeated until the algorithm reaches a maximum number of iterations or the energy is below a tolerance level.

(f) Compute the global energy value and update the source image. The process is resumed from the first step if all multiresolution levels have not yet been used.

In this case, the artificial diatoms have been generated using a registration method with 5 levels of resolution. Furthermore, to increase the amount of samples obtained, new images are also generated from one of the original images and the artificial image generated from it. This allows us to obtain several artificial images just from a source and a target image using the intermediate sample as source or target, as appropriate.

The method employed to assess the applied transformation is the normalized cross-correlation, which makes it possible to measure the differences between the source and the warped image.

The following steps have also been accomplished to generate the new diatom samples:

- Both images are padded and scaled to a size of 256 × 256 pixels.
- Registration is then applied and the resulting images are scaled to their original sizes.
- The dimensions of the intermediate warped image are scaled to a size between the target and source images. This new size depends on the normalized cross-correlation values between the warped and the source image (ncc_S) and between the warped and the target image (ncc_T). Cross-correlation values are compared following Eq. (21).

$$sim = \frac{ncc_T^2}{ncc_T^2 + ncc_S^2} \tag{21}$$

Therefore, when *sim* reaches values close to one, the warped image is scaled to a size close to that of the target image. Otherwise, the warped image maintains its dimensions close to the ones from the source.

Examples of the generated images are shown in Fig. 14.

4. B-Spline Composition and Level Sets Registration

Another registration approach used is a Free Form Deformation method (FFD) commonly used in non-rigid image registration (proposed in Chan et al. [33]). In FFD methods, the source image is usually embedded in a B-spline object which is then deformed. The spatial transformation of the image is described with the B-spline control points and the locations of these points are optimized to find the optimal transformation. However, the approach selected presents an alternative procedure where both the spatial transformation and the image are represented in terms of B-splines. The displacement field is then computed in each iteration of the registration procedure and the new update is composed using the previous transformation field.

To measure the differences between the target and the source image the method uses the sum of square differences (*SSD*). However, to avoid problems that could lead to non-smooth transformations, T, a regularization term is added:

$$E(T) = SSD(I_1, I_0) + \sigma_T Reg(T) \tag{22}$$

In addition, to facilitate the optimization, a hidden variable is added, and the energy function, E, is defined in terms of two transformations (C and T) and three terms:

$$E(C,T) = SSD(I_1, I_0 \circ C) + \sigma_x dist(T,C)^2 + \sigma_T Reg(T) \tag{23}$$

where $Reg(T)$ is a regularization term, σ_T is the amount of regularization, $dist(T,C) = ||T - C||$, and σ_x accounts for the spatial uncertainty between C and T.

In the method selected, the one provided in Chan et al. [34], both the maximum number of iterations and the tolerance

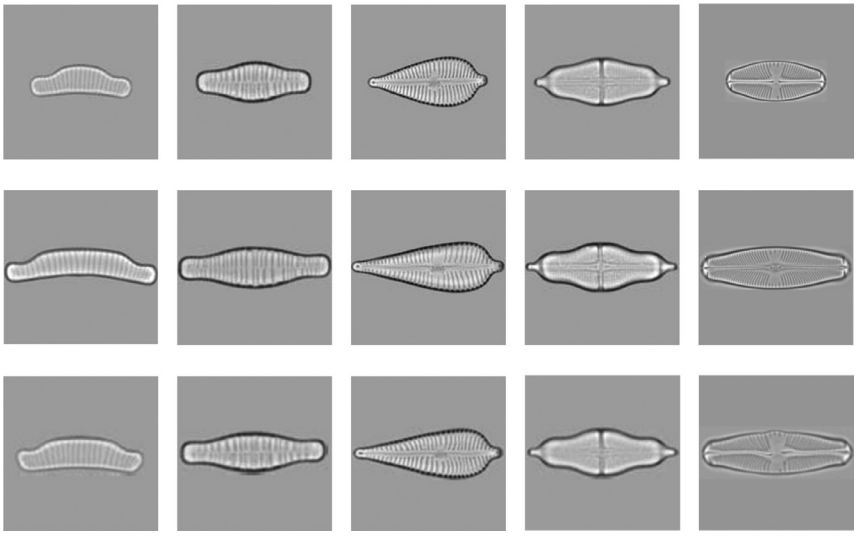


Fig. 14. Examples of registered diatoms. First row corresponds to the source image, second row to the target image, and the last row corresponds to the warped image.

value can be modified to stop the process. The tolerance parameter, *TOL*, allows us to stop the registration process for each sample when the differences between the target and the warped images are low enough. In this case, the difference between both samples, named residual value, is computed using the Euclidean norm as:

$$E_{norm}(I_1, I_0) = \sqrt{\sum_{k=1}^{N} |I_1(k) - I_0(k)|^2} \tag{24}$$

Therefore, the algorithm ends when one of the following conditions is fulfilled:

$$\frac{E_{norm}(I_1, I_0')}{E_{norm}(I_1, I_0)} < TOL \tag{25}$$

$$\frac{E^A_{norm} - E^B_{norm}}{3} < \frac{TOL}{0.05} \cdot 10^{-(4+nlevel-level)} \tag{26}$$

where I_0' is the updated source image, E^A_{norm} is the average of the last three residual values, E^B_{norm} is the average of the last three before the ones used to compute E^A_{norm}, *nlevel* is the number of multi-resolution levels, and *level* is the current one.

The maximum number of iterations used in the tests was 1000 to avoid lengthy computations, and three different tolerance values were used: 1%, 5% and 50%. Additionally, the registration process is repeated for each tolerance value. As a result, the method generates six new samples from each pair. Fig. 15 shows new diatom samples obtained with this method.

5. Matching CNNs for Registration

Matching CNNs mimic the classical registration process by combining the descriptors extracted from each image, resulting in a set of correspondences that are evaluated to compute the transformation to be applied. In the case of CNNs, this process is replicated using neural networks. The features that define each image can be extracted using a CNN and then another network can estimate the spatial transformation needed to perform the matching. In this work we use the method described in Rocco et al. [35], where the authors propose a neural network architecture organized in three stages (Fig. 16). These three stages are:

(a) Feature extraction nets
This first stage is composed of two identical VGG-16 networks trained with ImageNet without their fully connected layers at the end. The output of both networks are three dimensional matrices which represent the feature maps (or descriptors) of the introduced images.

(b) Matching network
Both feature maps are combined in this step to create a tensor that will be used in the subsequent regression stage. For each spatial location, this tensor contains all the similarity values between a specified feature in one image and all the features in the other.

(c) Regression network
Finally, the regression neural network computes the transformation applied to the source image.

Even though the method is already prepared for registration tasks, the neural network provided [36] has been retrained for the current work. The original diatom dataset includes 780 training and 97 validation samples. A simple data augmentation process is used to increase the number of images in both groups. For each sample, a spatial transformation and a gaussian smoothing filter with different sigma values are applied. Thus, 6 samples are obtained for each image, increasing the sizes of the training and validation datasets.

The architecture is trainable end-to-end. Thus, both the regression and feature extraction stages can be trained simultaneously. On the contrary, matching layers do not require training since they do not contain parameters to update.

The resulting network was then used to generate new diatom samples using every possible pair of samples in each diatom species. As in the other methods, the smallest diatom of the pair is used as the source image while the other one is the target image.

Both images are resized to 227×227 pixels since this is the input size of the VGG-16 network. In addition, the source image is padded before being used as input to prevent the warped image going outside the image limits. In this case, only one artificial sample is generated from each pair of diatoms. Some generated examples are shown in Fig. 17.

2.3. Invalid sample filtering

Since the data augmentation process applies some transformations that can deform the diatom contour or change the internal striae pattern leading to teratological samples with unnatural features, a process to distinguish between valid and invalid samples may be required. This task can be seen as an image quality as-

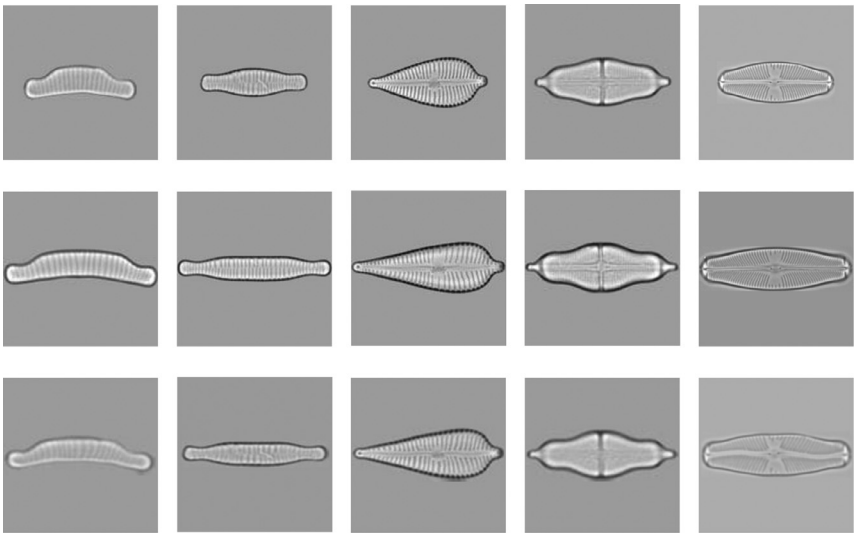


Fig. 15. Example of data augmentation using B-Spline registration. The first row corresponds to the source image, the second row to the target image and the third row to the warped image obtained.

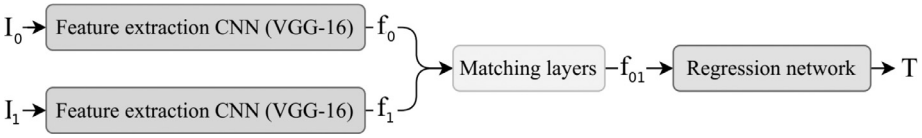


Fig. 16. CNN architecture proposed in Rocco et al. [35].

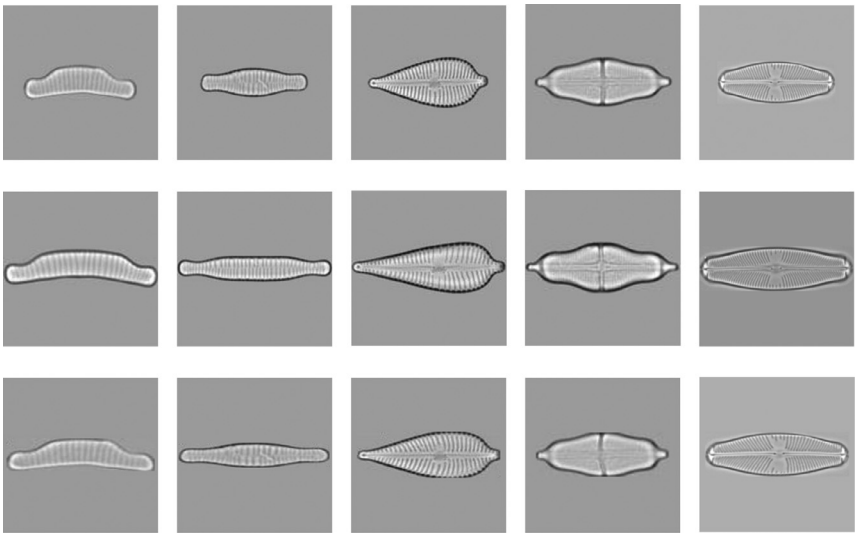


Fig. 17. Example of data augmentation using registration through CNNs. The first row represents the source images, the second row the target images, and the third row the generated images.

sessment (IQA) process in which a method is used to measure the quality of the images in terms of their similarity with the valid samples. In this work, three quality metrics have been computed to extract features from each pair composed of one of the source samples and one of the artificial samples: Visual Information Fidelity (VIF), Mean Structural Similarity (SSIM) and Singular Value Decomposition (SVD). The process is performed twice since there are two source images related to each artificial sample. The metric values computed are then used to represent sample realism.

This step is not carried out for glomeruli and pollen datasets since the images obtained in these cases do not present unnatural features. Images from the pollen dataset have similar shapes so that the resulting artificial samples do not differ much from the original ones. In case of glomeruli, they already have different shapes since it depends on where the cut of the specimen is located and how it is placed on the slide (pathology samples usually present some degree of distortion).

2.3.1. Visual information fidelity (VIF)

The visual information fidelity (VIF) metric is described in Sheikh and Bovik [37]. This method is based on modeling features captured by the human visual system (HVS) and the calculation of two mutual information measures. Following this approach both original and artificial diatom images are interpreted as signals which pass through the human HVS channel that interprets them before reaching the brain. However, the artificial diatom im-

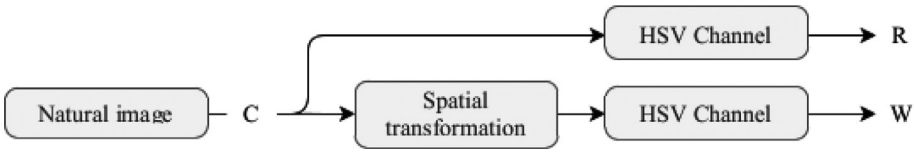


Fig. 18. VIF metric diagram. *C* is one of the samples from the original test set used to obtain the artificial sample. The artificial samples are considered as the output of a certain spatial transformation.

age is considered to have passed through a distortion channel before reaching the HVS channel.

Therefore, three signals can be distinguished (Fig. 18) and mutual information is calculated between signals *C* and *R* for the original image, and between *C* and *W* for the artificial image. Then, these two quantities are used to determine the level of visual similarity.

The algorithm employed follows a multiscale-based methodology applied in four iterations. In each iteration, the dimensions of both images are scaled to half their size, beginning at their original size, and the following parameters being computed:

$$\mu_1 = H(I_O) \tag{27}$$

$$\mu_2 = H(I_A) \tag{28}$$

$$\sigma_{\mu_1} = H(I_O \cdot I_O) - \mu_1^2 \tag{29}$$

$$\sigma_{\mu_2} = H(I_A \cdot I_A) - \mu_2^2 \tag{30}$$

$$\sigma_{\mu_1\mu_2} = H(I_O \cdot I_A) - \mu_1\mu_2 \tag{31}$$

$$g = \frac{\sigma_{\mu_1\mu_2}}{\sigma_{\mu_1} + \varepsilon} \tag{32}$$

$$sv = \sigma_{\mu_1} - g \cdot \sigma_{\mu_1\mu_2} \tag{33}$$

$$num = num + \sum log_{10}(1 + \frac{g_i^2 \sigma_{\mu_1}}{sv_i^2 + 2}) \tag{34}$$

$$den = den + \sum log_{10}(1 + \frac{\sigma_{\mu_1}}{2}) \tag{35}$$

where I_O and I_A are the original and the artificial images respectively, the $H(I)$ function represents a Gaussian filter applied in I, and *num* and *den* values are updated each iteration, setting their initial value to 0.

Finally, the *VIF* metric is computed as shown in Eq. (36). As a result, the similarity VIF metric is a value between 0 and 1, where a value of 1 means that both images are equal.

$$VIF = \frac{num}{den} \tag{36}$$

2.3.2. Mean structural similarity index measure(SSIM)

The structural similarity index measure (*SSIM*) is a perception-based method which measures the similarity between two images. This method considers image degradation as changes in the structural information. The comparison between the images is achieved through three main parameters: luminance (l), contrast (c) and structure (s), defined as follows:

$$l(x,y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \tag{37}$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \tag{38}$$

$$s(x,y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3} \tag{39}$$

where μ_x, μ_y, σ_x and σ_y correspond to the mean and variance for each image x and y, while c_1, c_2 and c_3 are regularization variables for low value denominators.

Finally, *SSIM* is computed as:

$$SSIM(x,y) = \left[l(x,y)^\alpha \cdot c(x,y)^\beta \cdot l(x,y)^\gamma\right] \tag{40}$$

where $\alpha > 0$, $\beta > 0$ and $\gamma > 0$ are parameters to adjust the importance of each component.

The output value is constrained between 0 and 1, where a value equal to 1 means that both images are the same, and 0 means that they are completely different.

2.3.3. Singular value decomposition (SVD)

The singular value decomposition (SVD) image quality assurance method was proposed in Wang et al. [38]. It is based on the idea of decomposing the image into content-dependent and content-independent partitions. While the quality of the content-dependent part is measured by gradient and contrast similarities, the quality of the content-independent part is measured using a normalized peak signal-to-noise ratio (PSNR).

The algorithm is applied over an image pair composed of an original and an artificial image where the latter has been generated by applying one of the data augmentation methods with the former sample and another original sample. Each image is then treated as a matrix, X, and decomposed into three matrices through the SVD factorization as:

$$X = U \times \triangle \times V^T \tag{41}$$

Then, the decomposition is used to obtain the images of the content-dependent part, R, and the least content-dependent (or content-independent) part, I, knowing that the structural component is dominated by the first basis images of the decomposition.

As a result, the gradient similarity $g(x,y)$ between the content-dependent images obtained from the original, R_O, and the artificial, R_A, inputs is defined as:

$$g(x,y) = \frac{2G_{R_O}(x)G_{R_A}(y) + k}{G_{R_O}(x)^2 G_{R_A}(y)^2 + k} \tag{42}$$

where $G_{R_O}(x)$ and $G_{R_A}(y)$ are the gradient values of the central pixels of image blocks x and y, and k is a constant to avoid dividing by 0. Four kernels are applied to compute the gradients in four directions and the maximum response over them is used as the magnitude value of the gradient.

The contrast similarity $c(x,y)$ is obtained with a similar procedure following:

$$g(x,y) = \frac{2\sigma_{R_O}(x)\sigma_{R_A}(y) + k}{\sigma_{R_O}(x)^2\sigma_{R_A}(y)^2 + k} \tag{43}$$

where $\sigma_{R_O}(x)$ and $\sigma_{R_A}(y)$ are the standard deviations of image blocks x and y.

(a) Source and target samples

(b) Sample generated from (a). VIF=(0.06,0.97), SSIM=(0.72,0.99), SVD=(20.02,0.16)

(c) Source and target samples

(d) Sample generated from (c). VIF=(0.01,0.28), SSIM=(0.78,0.91), SVD=(21.45,8.38)

Fig. 19. Examples of valid (b) and invalid (d) images generated with their metric values when they are compared with either the source or the target image (a) and (d).

Finally, the PSNR value is computed from I_O and I_A as:

$$PSNR(I_O, I_A) = \frac{1}{k} 10 \log_{10} \left(\frac{L^2}{MSE(I_O, I_A)} \right) \quad (44)$$

where MSE is the mean squared error and k is a constant used to normalize the PSNR value into [0,1].

2.4. Generated data and classifiers for invalid sample filtering

A total of 29,550 artificial diatom samples have been generated with 4000–8000 images per method. An expert diatomologist reviewed these images and labeled them as valid or invalid according to how realistic they are, i.e., if they maintain the morphological features of their species or instead, if their contour or striae has been altered substantially. The metrics were then computed for each sample generated. Fig. 19 shows the metric values for valid and invalid artificial diatoms.

With this data, classifiers such as support vector machines (SVM), linear and quadratic discriminant classifiers (LDC and QDC), k-nearest neighbor (kNN), and bagged trees (BT) were trained to discern between valid and invalid samples. The best results were obtained with the bagged trees classifier and are shown in Section 3.

2.5. Classification

After removing the wrong samples, 6 CNN architectures were trained to compare the classification results obtained with general data augmentation techniques and with the proposed data augmentation method. The networks employed are ResNet18, AlexNet, VGG11, SqueezeNet1.0, DenseNet121, and InceptionV3. All models were initialized with the ImageNet weights to follow a transfer learning procedure [39]. The number of epochs was set to 60 and the learning rate used was 0.001. All experiments were carried out on an NVIDIA Quadro P4000 graphic card with 8 GB of memory and were repeated 5 times.

On the dataset side, the data augmentation methods were only applied to the training samples. Thus, both validation and test sets underwent none of these data augmentation processes and were the same across both data augmentation approaches to provide a fair comparison.

Table 4
Percentage of images accepted and rejected by the expert through data augmentation method (DA): Stationary Velocity Field (SVF), Matching CNNs (MCNN), Diffeomorphic Log-Demons (DLD), B-Spline Composition and Level Sets (B-S&LS), and Morphing (MORPH).

DA method	% Accepted	% Rejected
SVF	86.60	16.40
MCNN	40.55	59.45
DLD	53.63	46.37
B-S&LD	41.93	58.07
MORPH	81.66	18.34

3. Results

3.1. Diatom dataset results

Before training the classification CNNs, an invalid image removal process have been carried out to check the influence of including or removing the generated teratological samples in the classifier performance and to know which of the proposed methods is able to generate more realistic samples.

3.1.1. Invalid image removal

After all generated samples were reviewed by the specialist, the number of both valid and invalid samples per group and data augmentation method employed were obtained (Fig. 20). The two methods with the best results are Stationary Velocity Field (SVF) and Morphing (MORPH) with 83.6% and 81.7% of valid samples generated respectively (Table 4). On the contrary, the Matching CNNs (MCNN) method obtains the worst results with only 40.6% of the generated samples being valid. Notice that this method requires retraining the networks with the dataset to improve the results but the diatom dataset is very small and this may lead to a performance drop.

By diatom class, *008-Nitzschia amphibia* was the most difficult to generate samples from, with all methods except SVF generating more invalid than valid samples. On the contrary, all methods generate more valid than invalid samples for *001-Eunotia tenella, 003-Gomphonema augur*, and *007-Nitzschia capitellata* classes.

With the results obtained by the specialist, a set of classifiers was trained with the metrics extracted from the images generated. These classifiers are: classification trees, linear and quadratic classifiers (LDC and QDC), naive Bayes classifier (NBC), support vector machines (SVM), k-nearest neighbors (kNN), boosted trees, and bagged trees. The training process follows a 5-fold cross-validation technique. When each of the metrics is applied individually, the accuracy obtained is 78.2% for VIF, 78.3% for MSS, and 76.4% for SVD. Combining two of the metrics increases the accuracy resulting in 79.1% of samples correctly classified for VIF + MSS, 80% for VIF + SVD, and 80.2% for MSS + SVD. Finally, when all metrics are used, results show an 81% accuracy with an area under the curve (AUC) of 0.89 using Bagged Trees (Fig. 21).

If the classifier threshold is adjusted, it is possible to filter 97% of the wrong samples at the cost of accepting only 50% of the re-

Fig. 20. Comparison between the number of accepted and rejected samples by diatom species and data augmentation method (Stationary Velocity Field (SVF), Matching CNNs (MCNN), Diffeomorphic Log-Demons (DLD), B-Spline Composition and Level Sets (B-S&LS), and Morphing (MORPH)).

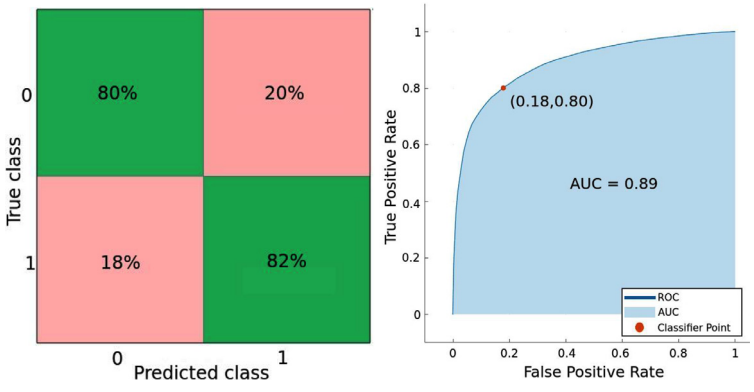


Fig. 21. Confusion matrix and ROC from Bagged Trees classifier trained with valid an invalid samples generated by Morphing and Stationary Velocity Field data augmentation methods.

Table 5
Number of images per set and data augmentation (DA) method for Diatom dataset (without data augmentation (w/o-DA), geometric transformations (Geo-DA), noise injection (Noise-DA), artificial image generation with GANs (GAN-DA), proposed method with invalid image filtering (P-DA (filter)) and proposed method without the filter (P-DA)).

#Train						#Val.	#Test
w/o-DA	Geo-DA	Noise-DA	GAN-DA	P-DA (filter)	P-DA		
780	46,800	46,800	28,780	15,532	29,550	97	99

Table 6
Diatom dataset average classification results per DA method (without data augmentation (w/o-DA), geometric transformations (Geo-DA), noise injection (Noise-DA), artificial image generation with GANs (GAN-DA), proposed method with invalid image filtering (P-DA (filter)) and proposed method without the filter (P-DA)).

DA	% Acc. validation	% Acc. test
w/o-DA	**99.79 ± 0.24**	98.82 ± 0.98
Geo-DA	99.24 ± 0.60	96.97 ± 0.95
Noise-DA	99.38 ± 0.17	98.05 ± 0.88
GAN-DA	99.69 ± 0.34	98.55 ± 0.71
P-DA (filter)	99.76 ± 0.15	99.26 ± 0.70
P-DA	99.38 ± 0.50	**99.29 ± 0.48**

alistic ones. However, with the two methods chosen, it is possible to generate an initial set of artificial samples using a larger number of images, while keeping in mind that some of them may be filtered.

3.1.2. Classification

A set of experiments was conducted varying the CNN architecture and data augmentation technique to automatically classify diatoms (see Section 2.5). All experiments used the same randomly selected train/validation/test partition and were run 5 times to obtain the average values. In this section both validation and test results are shown.

Table 5 shows the number of images per set and DA method: without data augmentation (w/o-DA), geometric transformations (Geo-DA), noise injection (Noise-DA), artificial image generation with GANs (GAN-DA), proposed method with invalid image filtering (P-DA (filter)) and proposed method without the filter (P-DA).

The best results were obtained with the P-DA methods without removing the invalid samples (Accuracy = 99.29% and STD = 0.48). The results obtained after applying the filter are very similar to the results of not applying it with a difference of only 0.03% of accuracy. Thus, the inclusion of teratological samples in the classifier training does not affect its performance. However, it is worth noting that the models trained with these unrealistic samples could be less robust to adversarial attacks as it has been shown in the literature [40,41].

The other three DA methods obtained worse results than training the classifier without employing DA. A possible cause for Geo-DA and Noise-DA is the addition of artifacts that are not present in the test distribution. On the other hand, when GAN-DA is used, the resulting images are all very similar, leading to training batches with less feature variability. In addition, this dataset is already composed of samples from different stages of the diatom life cycle, therefore, there is already some intrinsic variability in it (Table 6).

In most tests, the classes that are usually confused are *012-Sellaphora capitata* with *011-Sellaphora blackfordensis*. Fig. 22 shows images from these classes. These missclassications are caused by the great resemblance between these two species that are hard to distinguish even for the human eye.

(a) *Sellaphora capitata*

(b) *Sellaphora blackfordensis*

Fig. 22. Species that are sometimes confused due to their similarity.

Table 7
Average results by each DA method proposed filtering invalid samples (Stationary Velocity Field (SVF), Matching CNNs (MCNN), Diffeomorphic Log-Demons (DLD), B-Spline Composition and Level Sets (B-S&LS), and Morphing (MORPH)).

Network	% Acc. validation	% Acc. test
SVF	99.37 ± 0.63	98.77 ± 1.23
MCNN	**99.89 ± 0.33**	98.93 ± 0.88
DLD	99.77 ± 0.57	99.16 ± 0.79
B-S&LS	99.37 ± 0.52	**99.38 ± 0.86**
MORPH	99.71 ± 0.48	99.05 ± 1.07

When each of the DA methods proposed is applied individually, all except for Stationary Velocity Field (SVF) achieve average accuracies that outperform the results obtained with the rest of the methods (Table 7). The largest improvement is achieved by B-Spline Composition and Level Sets Registration (B-S) followed by Diffeomorphic Log-Demons Registration and Morphing.

3.2. Glomeruli and pollen datasets results

For the classificaton of Glomeruli and Pollen, only Morphing was applied for data augmentation. We have selected it since it is the fastest of the methods studied in this work (see Table 8), is able to obtain good results and generates realistic samples for the diatom dataset (see Table 4). The 6 CNN architectures selected were also trained and tested 5 times as done in the experiments with the diatom dataset. The number of samples employed in each case is shown in Table 9.

Table 8
Time, in seconds, to obtain an artificial image per DA method (Stationary Velocity Field (SVF), Matching CNNs (MCNN), Diffeomorphic Log-Demons (DLD), B-Spline Composition and Level Sets (B-S&LS), and Morphing (MORPH)).

Time	SVF	MCNN	DLD	B-S&LS	MORPH
Time	14.41	0.31	0.48	4.16	**0.02**

Table 9
Number of images per set and DA method for Glomeruli and Pollen datasets (without data augmentation (w/o-DA), geometric transformations (Geo-DA), noise injection (Noise-DA), artificial image generation with GANs (GAN-DA), proposed method with morphing (P-DA (MORPH))).

Dataset	#Train					#Val.	#Test
	w/o-DA	Geo-DA	Noise-DA	GAN-DA	P-DA (MORPH)		
Glomeruli	926	55,560	55,560	4926	21,878	200	200
Pollen	1703	102,180	102,180	151,703	38,910	445	445

Table 10
Gomeruli dataset average classification results per DA method (without data augmentation (w/o-DA), geometric transformations (Geo-DA), noise injection (Noise-DA), artificial image generation with GANs (GAN-DA), proposed method with morphing (P-DA (MORPH))).

DA	% Acc. validation	% Acc. test
w/o-DA	**97.95 ± 0.24**	98.33 ± 0.57
Geo-DA	97.87 ± 0.27	98.53 ± 0.40
Noise-DA	97.88 ± 0.32	98.32 ± 0.45
GAN-DA	94.57 ± 0.68	96.77 ± 1.08
P-DA (MORPH)	97.37 ± 0.38	**100 ± 0**

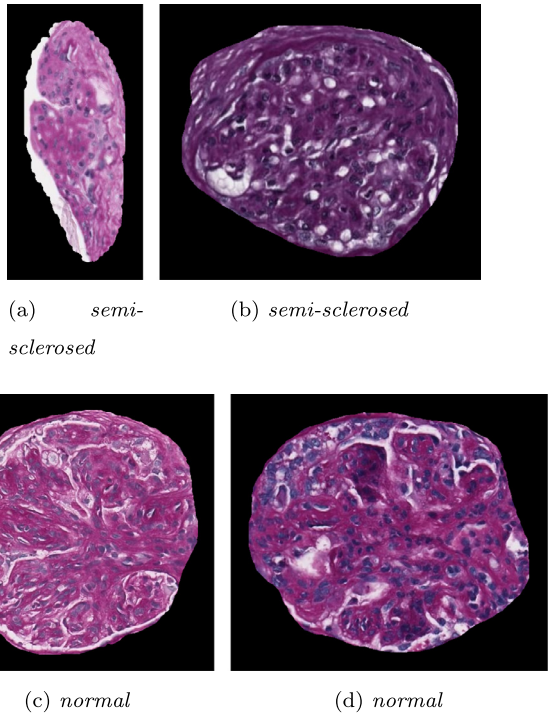
Results show an improvement of the proposed DA method based on morphing between 1.47% and 3.23% over the rest of the methods for the Glomeruli Dataset being able to correctly classify all test samples (Table 10). Glomeruli highly vary in texture and shape and these characteristics make the proposed method more suitable to solve the problem because it adds intermediate shapes and textures that are not already present in the dataset but can be found in other samples. In this case, texture plays an important role in the identification of a sclerosed or semi-sclerosed glomerulus.

In this case, G-DA obtains better results than not using DA, Noise-DA obtains similar results and GAN-DA obtains the worst scores. This is caused by a problem called "model collapse" which occurs when the generator finds a point in the data distribution where the discriminator fails and keeps generating images around this point. Since the dataset is small, the GAN is not able to find other points in the dataset distribution to generate artificial samples from them and trick the discriminator.

Fig. 23 shows images that were correctly classified by our approach but not by the others.

For the Pollen dataset, results show an improvement between 0.23% and 2.16% over the rest of the methods when Morphing is applied for DA (with an average accuracy of 91.68% and a STD of 0.46) (Table 11). In this case, both G-DA and Noise-DA outperforms not using any DA technique because there is some noise already present in the dataset and rotations can be seen in the dataset samples. Although the proposed DA method obtains the highest scores, the difference with other techniques is not as large as in the Glomeruli task due to the high similarity of the samples that belong to the same class. This leads us to think that the proposed method works better when samples show larger variability in their shape and texture. In fact, examining the different classes of pollen, accuracy increases the most for the classes with larger variability in shape and texture (Fig. 24).

Fig. 24 shows images that were correctly classified by our approach but not by the others.

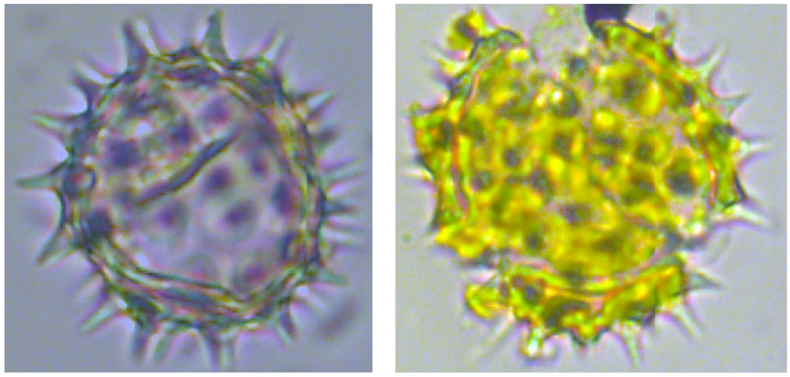


(a) *semi-sclerosed* (b) *semi-sclerosed* (c) *normal* (d) *normal*

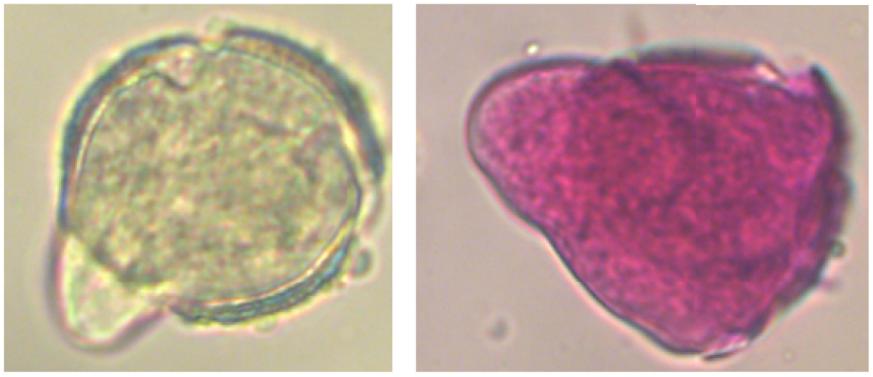
Fig. 23. Examples of glomeruli that are correctly classified by our DA method but not by the others. Notice the shape difference of the gomerulus in a) and the similarity in texture between glomeruli b) and d).

Table 11
Pollen dataset average classification results per DA method (without data augmentation (w/o-DA), geometric transformations (Geo-DA), noise injection (Noise-DA), artificial image generation with GANs (GAN-DA), proposed method with morphing (P-DA (MORPH))).

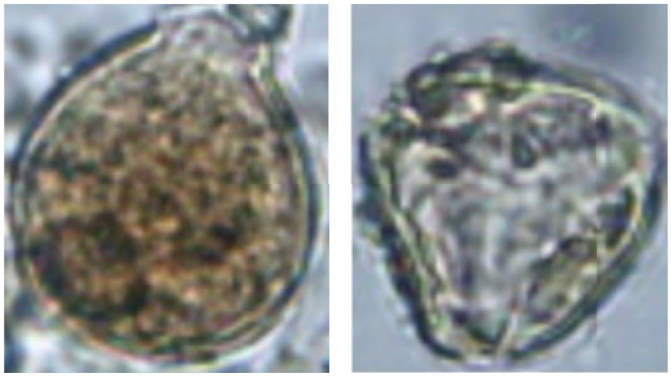
DA	% Acc. Validation	% Acc. Test
w/o-DA	91.64 ± 0.36	90.42 ± 0.75
Geo-DA	92.13 ± 0.57	91.36 ± 0.51
Noise-DA	**92.95 ± 0.49**	91.45 ± 0.98
GAN-DA	91.19 ± 0.55	89.52 ± 0.81
P-DA (MORPH)	92.33 ± 0.40	**91.68 ± 0.46**



(a) *aspilia grazielae* (b) *aspilia grazielae*



(c) *tabebuia rosealba* (d) *tabebuia rosealba*



(e) *dipteryx alata* (f) *dipteryx alata*

Fig. 24. a), c) and e) are examples of pollen samples that are correctly classified by our DA method but not by the others. b), d) and f) are training samples from the same classes that shows the variability in shape, color and texture.

4. Conclusions

This work addresses the problem of automatically classifying objects with highly variable shape and texture using CNNs on extremely small datasets.Given the difficulty of acquiring and labeling additional taxa, particularly in problems that require expert labeling, a novel data augmentation method based on Morphing and Registration is proposed. The method has been applied to three datasets representing three distinct scenarios: diatom, pollen and glomeruli identification, obtaining results that outperform traditional methods for increasing dataset size. This was demonstrated by training and comparing six different CNN architectures with the proposed DA method and with general DA methods. The proposed technique improves accuracy by 0.47%, 1.47%, and 0.23% over existing techniques for diatom, glomeruli and pollen problems, respectively.

For the Diatom dataset, the method is capable of simulating the shape changes associated with various stages of the diatom life cycle, resulting in images that resemble newly acquired samples with intermediate shapes. Indeed, the other methods compared produced results that were inferior to those obtained without data augmentation. For the Glomeruli dataset, the method is capable of adding new samples with varying shapes and degrees of sclerosis (through different textures). This is the case where our

proposed DA method is more beneficial, when objects highly differ in both shape and texture. Finally, for the Pollen dataset, our method still improves the results, despite the fact that there are only minor variations between samples in a few classes and the dataset contains additional features such as noise that are likely to benefit other existing DA techniques.

Finally, to discard artificial diatoms with teratologies, a classifier was trained using bagging trees. The input variables of the bagging trees were the VIF, SSIM and SVD quality metrics calculated between the two original images (source and target images) and the synthetic image generated from them. The classifier achieves an AUC of 0.86 although with threshold adjustment it is possible to obtain only 3% of false positive samples, and therefore an artificial dataset with 97% of valid samples. While removing or keeping these samples had no effect on the classifiers' performance, it should be considered if the model's robustness to adversarial attacks is critical.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Noelia Vallez: Conceptualization, Data curation, Formal analysis, Writing – original draft, Writing – review & editing, Validation. **Gloria Bueno:** Conceptualization, Data curation, Formal analysis, Writing – original draft, Writing – review & editing, Validation. **Oscar Deniz:** Conceptualization, Data curation, Formal analysis, Writing – original draft, Writing – review & editing, Validation. **Saul Blanco:** Conceptualization, Data curation, Formal analysis, Writing – original draft, Writing – review & editing, Validation.

Acknowledgements

The authors acknowledge financial support of the Spanish Government and Junta de Comunidades de Castilla-La Mancha under projects AQUALITAS (Ref. CTM2014-51907-C2-R-MINECO), HYPERDEEP (Ref. SBPLY/19/180501/000273), and APRENDAMOS (Ref. SBPLY/17/180501/000543). They would also like to extend the acknowledgment to technicians Enrique Cepeda and Jesus Diaz for their help in running some experiments.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.cmpb.2022.106775.

References

[1] A. Poulickova, D.G. Mann, D.G. Mann, Diatom Sexual Reproduction and Life Cycles, John Wiley & Sons, Ltd, 2019, pp. 245–272, doi:10.1002/9781119370741.ch11.
[2] A. Kale, B. Karthick, The diatoms: big significance of tiny glass houses, Resonance 20 (2015) 919–930.
[3] G. Cristobal, S. Blanco, G. Bueno, Modern Trends in Diatom Identification Fundamentals and Applications: Fundamentals and Applications, Springer International Publishing, 2020, doi:10.1007/978-3-030-39212-3.
[4] A. Pedraza, G. Bueno, O. Deniz, G. Cristóbal, S. Blanco, M. Borrego-Ramos, Automated diatom classification (Part B): a deep learning approach, Appl. Sci. 7 (5) (2017) 1–25. 460.
[5] C. Sanchez-Bueno, G. Cristobal, G. Bueno, Diatom identification including life cycle stages through morphological and texture, PeerJ Life Environ. 7 (2019) e6770.
[6] C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning, J. Big Data 6 (2019) 60.
[7] D. Su, H. Kong, Y. Qiao, S. Sukkarieh, Data augmentation for deep learning based semantic segmentation and crop-weed classification in agricultural robotics, Comput. Electron. Agric. 190 (2021) 106418.
[8] M.E. Akbiyik, Data augmentation in training {CNN}s: injecting noise to images, in: ICLR 2020 Conference, 2020, pp. 1–9. https://openreview.net/forum?id=SkeKtyHYPS.
[9] M.H. Yap, M. Goyal, F. Osman, R. Martí, E. Denton, A. Juette, R. Zwiggelaar, Breast ultrasound region of interest detection and lesion localisation, Artif. Intell. Med. 107 (2020) 101880.
[10] Z. Zhong, L. Zheng, G. Kang, S. Li, Y. Yang, Random erasing data augmentation, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, 2020, pp. 13001–13008.
[11] R. Takahashi, T. Matsubara, K. Uehara, Data augmentation using random image cropping and patching for deep CNNs, IEEE Trans. Circuits Syst. Video Technol. 30 (2019) 2917–2931.
[12] V. Sandfort, K. Yan, P.J. Pickhardt, R.M. Summers, Data augmentation using generative adversarial networks (cyclegan) to improve generalizability in CT segmentation tasks, Sci. Rep. 9 (2019) 1–9.
[13] J.P. Cohen, M. Luck, S. Honari, Distribution matching losses can hallucinate features in medical image translation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2018, pp. 529–536.
[14] U. Scherhag, C. Rathgeb, J. Merkle, R. Breithaupt, C. Busch, Face recognition systems under morphing attacks: asurvey, IEEE Access PP (2019) 1.
[15] S. Hauberg, O. Freifeld, A.B.L. Larsen, J. Fisher, L. Hansen, Dreaming more data: class-dependent distributions over diffeomorphisms for learned data augmentation, in: Artificial Intelligence and Statistics, PMLR, 2016, pp. 342–350.
[16] M. Orbes-Arteaga, L. Sørensen, J. Cardoso, M. Modat, S. Ourselin, S. Sommer, M. Nielsen, C. Igel, A. Pai, Paddit: probabilistic augmentation of data using diffeomorphic image transformation, in: Medical Imaging 2019: Image Processing, vol. 10949, SPIE, 2019, pp. 197–202.
[17] L. Nanni, M. Paci, S. Brahnam, A. Lumini, Comparison of different image data augmentation approaches, J. Imaging 7 (2021) 254.
[18] G. Bueno, M.M. Fernandez-Carrobles, L. Gonzalez-Lopez, O. Deniz, Glomerulosclerosis identification in whole slide images using semantic segmentation, Comput. Methods Prog. Biomed. 184 (2020) 105273.
[19] S. Blanco, Diatom life cycle images dataset, 2018, 10.6084/m9.figshare.7077725.
[20] D. Mann, M. Bayer, Diatom size reduction image sets for shape and appearance models, 2018, http://rbg-web2.rbge.org.uk/DIADIST/.
[21] D.G. Mann, S.M. McDonald, M.M. Bayer, S.J. Droop, V.A. Chepurnov, R.E. Loke, A. Ciobanu, J.H.D. Buf, The Sellaphora pupula species complex (Bacillariophyceae): morphometric analysis, ultrastructure and mating data provide evidence for five new species, Phycologia 43 (2004) 459–482.
[22] A.B. Goncalves, J. Souza, G. Silva, M. Cereda, A. Pott, M. Naka, H. Pistori, Feature extraction and machine learning for the classification of brazilian savannah pollen grains, PLoS One 11 (2016) e0157044.
[23] R. Redondo, G. Bueno, F. Chung, R. Nava, V. Marcos, G. Cristobal, T. Rodriguez, A. Gonzalez-Porto, C. Pardo, O. Deniz, B. Escalante-Ramirez, Pollen segmentation and feature evaluation for automatic classification in bright-field microscopy, Comput. Electron. Agric. 110 (2015) 56–69.
[24] G. Astolfi, A.B. Gonçalves, G.V. Menezes, F.S.B. Borges, A.C.M.N. Astolfi, E.T. Matsubara, M. Alvarez, H. Pistori, Pollen73s: an image dataset for pollen grains classification, Ecol. Inform. 60 (2020) 101165.
[25] X. Wang, K. Wang, S. Lian, A survey on face data augmentation, CoRR abs/1904.11685(2019).
[26] F.P. Kuhl, C.R. Giardina, Elliptic Fourier features of a closed contour, Comput. Graph. Image Process. 18 (1982) 236–258.
[27] M. Niethammer, R. Kwitt, F. Vialard, Metric learning for image registration, CoRR abs/1904.09524(2019).
[28] Z. Shen, X. Han, Z. Xu, M. Niethammer, Networks for joint affine and non-parametric image registration, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 4224–4233.
[29] M. Niethammer, Z. Shen, R. Kwitt, Mermaid: image registration via autoMAtic differentiation, 2019, (https://github.com/uncbiag/registration). Accessed: 2022-02-28.
[30] J.P. Thirion, Image matching as a diffusion process: an analogy with Maxwell's demons, Med. Image Anal. 2 (1998) 243–260.
[31] T. Vercauteren, X. Pennec, A. Perchant, N. Ayache, Symmetric log-domain diffeomorphic registration: ademons-based approach, in: Medical Image Computing and Computer Assisted Intervention – MICCAI 2008, Springer Berlin Heidelberg, 2008, pp. 754–761, doi:10.1007/978-3-540-85988-8_90.
[32] H. Lombaert, Diffeomorphic log demons image registration, 2014, (https://www.mathworks.com/matlabcentral/fileexchange/39194-diffeomorphic-log-demons-image-registration). Accessed: 2021-09-01.
[33] C.L. Chan, C. Anitescu, Y. Zhang, T. Rabczuk, Two and three dimensional image registration based on B-spline composition and level sets, Commun. Comput. Phys. 21 (2017) 600–622.
[34] C.L. Chan, C. Anitescu, Y. Zhang, T. Rabczuk, 2D and 3Dspline-based image registration, 2020, (https://github.com/stellaccl/cdmffd-image-registration). Accessed: 2022-02-28.
[35] I. Rocco, R. Arandjelovic, J. Sivic, Convolutional neural network architecture for geometric matching, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 39–48.
[36] I. Rocco, R. Arandjelovic, J. Sivic, Convolutional neural network architecture for geometric matching, 2017b, (https://github.com/ignacio-rocco/cngeometric_matconvnetb). Accessed: 2022-02-28.
[37] H.R. Sheikh, A.C. Bovik, Image information and visual quality, IEEE Trans. Image Process. 15 (2006) 430–444.

[38] S. Wang, D. Cui, B. Wang, B. Zhao, J. Yang, A perceptual image quality assessment metric using singular value decomposition, Circuits, Syst., Signal Process. 34 (2015) 209–229.
[39] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (2010) 1345–1359.
[40] A. Pedraza, O. Deniz, G. Bueno, On the relationship between generalization and robustness to adversarial examples, Symmetry 13 (2021) 817.
[41] H. Eghbal-zadeh, K. Koutini, P. Primus, V. Haunschmid, M. Lewandowski, W. Zellinger, B.A. Moser, G. Widmer, On data augmentation and adversarial risk: an empirical analysis, arXiv:2007.02650(2020).