# High Fidelity Visualization of What Your Self-Supervised Representation Knows About

**Anton Bothin** **Ioannis Iakovidis** **Giovanni Castellano**

## Abstract

In this report we present our attempt to reproduce the paper "High Fidelity Visualization of What Your Self-Supervised Representation Knows About" [1]. This paper aims to help researchers better understand what their image-based models learn by providing a way to visualize what information the representations they create contain. This is achieved by training a Representation Conditional Diffusion Model (RCDM) to generate images similar to those the original model is trained on while conditioning its output by that model's representation. We trained a number of RCDMs based on the representations of both supervised and self-supervised models and compared the images they generate to understand exactly what each model has learnt.

## 1 Introduction

Although neural networks are one of the most well-studied and widely applied areas of machine learning, they are notoriously difficult to understand due to their "black box" nature. In order to solve the understandability problem of image-based deep networks, the paper we are reproducing ("High Fidelity Visualization of What Your Self-Supervised Representation Knows About" [1]) attempts to directly visualize the information learnt by image-based models by directly visualizing it using a Representation-Conditioned Diffusion Model, a type of generative model. For each layer of an image-based deep network, the values the neurons of that layer take when fed a specific image can be understood as a representation of that image from the layer. By conditioning the image generation process of an RCDM with such a representation, the diffusion model produces images that represent the information the original model has stored in that representation. By repeatedly generating images from the same representation and observing what aspects of the image remain consistent we can observe what information the representation contains.

The paper is particularly interested in comparing the information contained in the representations of supervised versus self-supervised models. Self-supervised models are models that are trained on unlabeled data, usually by training the model to produce similar representations for copies of an image that have been subjected to different augmentations.

In order to understand the differences in information contained in supervised and self-supervised image models the authors of the original paper trained Representation-Conditioned Diffusion Models for four self-supervised and one supervised model trained on the Imagenet dataset and compared the images produced by their respective representations of images in the validation part of the Imagenet dataset. For the self-supervised models, images were produced using as conditioning the models' representation from both the backbone and the projector layers. All diffusion models were able to produce images similar to those whose representation they were conditioned with, in both in-sample and out-of-sample scenarios. Based on the images produced, it was revealed that the representations from the supervised model and the backbone of the self-supervised models contain not only information about the class of the object but also its context (size, background and so on). On the other hand the representations contained in the head of the self-supervised models lacked this extra context, resulting in the generation of more varied images. Images were also produced based

Project report for DD2412 or FDD3412

on the representations of variously augmented images to check whether those representations are invariant to the augmentations. The results indicate that supervised and backbone-level unsupervised representations contain information on the scale of the object, as well as the color and the background of the image. On the other hand, the representations from the head of the self-supervised models contain information only on the scale of the objects. Moreover, series of images were produced by interpolating between the representations of two different images, resulting in images containing elements of both classes. Generating images from representations obtained from adversarial images showed that self-supervised models are much more resilient to adversarial attacks than supervised ones. Finally, by comparing the representations of images with those of similar images and by generating images based on the differences of the representations the paper showed that information about specific aspects of an image can be extracted from specific dimensions of it's representation.

Given our limited computational resources we trained a number of small-sized RCDMs that generate 32x32 size images and generated a number of images replicating most of the experiments mentioned above. Although we were able to confirm many of the conclusions of the original paper, we observed some different results due to our different setup and dataset.

## 2 Code repository

`https://github.com/jupiter24/deep-learning-advanced-course`

## 3 Related Work

Denoising Diffusion Probabilistic Models [2] are a family of generative models that have achieved great success in image generation. These models are designed to take as input an image of random noise and generate as output an image that resembles images of the dataset the diffusion model was trained on. This task is accomplished by subtracting small amounts of noise from the input image over hundreds of steps until it produces the desired output. Their training is accomplished by taking images, adding small amounts of noise to them over hundreds of steps until they resemble random noise, and training the diffusion model to reverse this process step-by-step.

Self-supervised learning refers to a family of model-training methods that seek to exploit the vast amounts of unlabeled data available for many tasks. In order to use the unlabeled data for training, they are usually perturbed in some way and the model is tasked with either recognising the exact perturbation or reconstructing the original data. Examples of such pretext tasks on images include recognising the rotation of images, re-colorizing grayscale versions of images or filling in masked areas of images. A more modern approach for self-supervised learning on image-based models is to feed the model two versions of the same image, each with different augmentations, and train the representations the model produces to be as similar as possible[3]. After the self-supervised training is finished, the final layers of the model (called head or projector) are usually discarded and a new head is placed on top of the remaining model (backbone) and trained on the final task of the model. This procedure is used due to the fact that deep networks tend to store more general information in their earlier layers and more task-specific information in their latest ones.

## 4 Data

The model was trained using the ImageNet dataset, specifically the ILSVRC (ImageNet Large Scale Visual Recognition Challenge) subset. However, training an RCDM is a computationally intensive task, with the previous papers reporting 54 GPU days for training similar models on ImageNet with 128x128 size images[4]. For this project, such resources were not available. Thus, the training process had to be altered in order to make training several models feasible. Therefore, a subset of 50 classes were selected. Further, the images were resized to 32x32. We had a long debate on how to choose the 50 classes. On one hand, by choosing diverse classes we would get a dataset with many different images and therefore the generative model would have more information about the representation space. However, the representations of images belonging to different classes will live far from each other in the representation space, which means that the subset of the representation space we are going to study is relatively big. If we take 50 similar classes instead (for example 50 classes of animals), all the samples will live close to each other in the representation space, which

means that we are focusing our study on a smaller subset of the representation space, and it should be easier for the generative model to generate the samples since the training set contains many similar images.

We decided to perform most of our experiments using 50 randomly picked classes, however, we also tried to train one of the models on 50 similar classes (the first 50 classes of the dataset, which contain animal classes, with some of them being very similar to each other) to compare the results.

## 5 Methods

As mentioned in Section 1, this project has implemented RCDM [1], a model that can be used for visualizing representations learned by self-supervised and supervised models. The RCDM architecture is based on the Ablated Diffusion Model (ADM) [4], which in turn uses a U-Net architecture. The U-Net consists of a series of convolutional layers, each followed by a non-linear activation function. These convolutional layers reduce the spatial dimensions of the input image while increasing the number of feature channels, which captures more and more abstract features as the image is processed. The contracting path is then followed by a series of up-sampling layers, which increase the spatial dimensions while decreasing the number of feature channels. This allows the model to combine the high-level, abstract features from the contracting path with the spatial information from the input image to generate a high-resolution output image.

In order to condition the RCDM with the input representation, the original paper used a technique called conditional batch normalization. This technique normalizes the activation of each layer in a mini-batch based on a given condition, such as a representation. In theory this should help the U-Net focus on more important features of a given image, which in turn should improve the stability of the training process. However, another approach can also be taken, which is to add the representations to the conditioning used for determining which denoising step the model should perform. The original paper which our work is based on did not find any differences in term of experimental results between these two methods. Because of this, we chose to utilize the second approach due to its simplicity to implement.

Since the original paper extended the code of the original ADM paper [4], this project did the same. The original code for that paper can be found at `https://github.com/openai/guided-diffusion`. However, that paper trained exclusively on high resolution images and had tuned their hyperparameters accordingly. Because of this, we used hyperparameters which had been found in the paper "Improved Denoising Diffusion Probabilistic Model" [5], which had trained on the CIFAR-10 dataset which consists of 32x32 images, similar to our dataset. No hyperparameter tuning was performed for our particular task due to time constraints.

In our experiments, training was done on representations from two different pretrained models: (1) a self-supervised model called DINO [6], using the ResNet50 architecture; and (2) a supervised ResNet50 model. In order to keep the focus on small-sized images, these models were also given as input the downsampled 32x32 images. For the DINO model, three different RCDMs were trained using representations from three different stages of the architecture. Representations were taken from the projector, the backbone, and from the last downsampling layer in the backbone. The use of this last representation is an extension of the original paper's experiments and allows us to discover what information is encoded only on the last layers of the backbone. However the results from this model did not differ significantly from those of the backbone so they are omitted from this report.

## 6 Experiments and findings

To train the RCDM efficiently, Google Cloud was utilized. The training was performed on a deep learning virtual machine with a T4 GPU. This provided the necessary computational power to train the model in the 2 GPU days reported in Section 4. For storage, a SSD persistent disk with 350 GB of memory was used, which allowed us to store the ImageNet dataset. The models were each trained until the loss plateaued, at about 6000 steps. Although further improvement was detected in models we trained for longer, this required much more training time for a very small improvement so in the interest of time the training wasn't continued for the models used in the experiments. After training the models, we tried to replicate the experiments of the paper in order to study the structure of the representation spaces.

### 6.1 Data Augmentation

Augmenting the input data is one of the most interesting experiments we can perform to compare how the three models (supervised, Dino backbone, and Dino head) map the input data to the representation space. Figure 1 shows the results we obtained with the supervised model. We notice that in general the quality of the generated samples is not very good. The RCDMs for instance are not able to generate the car and also fail to generate the small lemon. However, if we increase the size of the lemon the model is able to generate good samples. Also, the samples of the bigger car are better compared with the samples of the smaller one. In both cases, the model is not able to generate the car, but if we increase the size of the object we can notice that the colors of the generated image are similar to the colors of the car. With the small car instead, the model seems to focus on the background rather than the object. We can conclude that objects with bigger sizes are easier to generate. Moreover, from the duck images, we can understand that the representations contain information about the background. These results are probably due to the fact that the classification models perform worse (and therefore provide worse representations) when given as input lower-resolution images than what they were trained on[7]. Another reason for this performance could be the fact that we stopped the model training before achieving the minimum possible loss.

Regarding the grayscale and color jitter augmentations, our results show that the representation contains information about the colors of the image. However, after changing the colors it becomes harder for the model to generate good samples. Despite the poor quality of the samples due to the small size of the training set, these results are still remarkable: the generative model is able to extract information about the size and the color of the object from the representation, without having been trained on such augmentations. This proves that the representation space is structured in such a way as to make it possible for the RCDM to infer this information.

Figure 2 and 3 show the same experiments performed using the Dino backbone and the Dino head representations. The results are very similar to the previous ones. We can observe with the grayscale and color jitter augmentations that the backbone representation space contains information about colors, in fact, the color of the samples changes as we augment the input data (figure 2 right). The representation space after projection (head) instead, seems to not contain as much information about colors and background(figure 3 right). This is consistent with one of the main findings of the original paper which states that the representation after projection is invariant to some augmentations and contains less info about the global context of the image (see figure 25 of the original paper [1]). Finally, figure 4 shows the results obtained training using a different training set. As mentioned in section 4 we thought it could be interesting to study how the results change as we focus on a smaller subset of the representation space (more similar training images). We can see that in general the generated samples seems slightly better compared to figure 2, but we do not observe any significant difference.

### 6.2 Interpolation

Generating samples conditioning on the interpolation of the representations of two different images provides other information about the structure of the representation space. As we expected, figure 5 shows that as we move in the representation space from one image to another one, the generated samples change accordingly. Also in this case, it is important to mention that the generative models were not trained on interpolated images.

### 6.3 Adversarial Attack

The idea behind this experiment is to perform an adversarial attack in order to increase the loss function of the classification model on a given image, and afterward, to observe the samples generated conditioning on the adversarial representation. Due to limited time and resources, we could not train our own classifiers, instead, we used the pre-trained classifiers provided by the Pytorch library and the original Dino paper [6]. However, such classifiers were trained on the whole Imagenet dataset (1000 classes), whereas our generative models are only trained on a subset of 50 classes. This means that it is very likely that the adversarial attack moves the representation of the attacked image towards a representation corresponding to a class our generative model is not able to generate. The results are illustrated in figure 6. We can see that as we increase the strength of the attack the quality of the

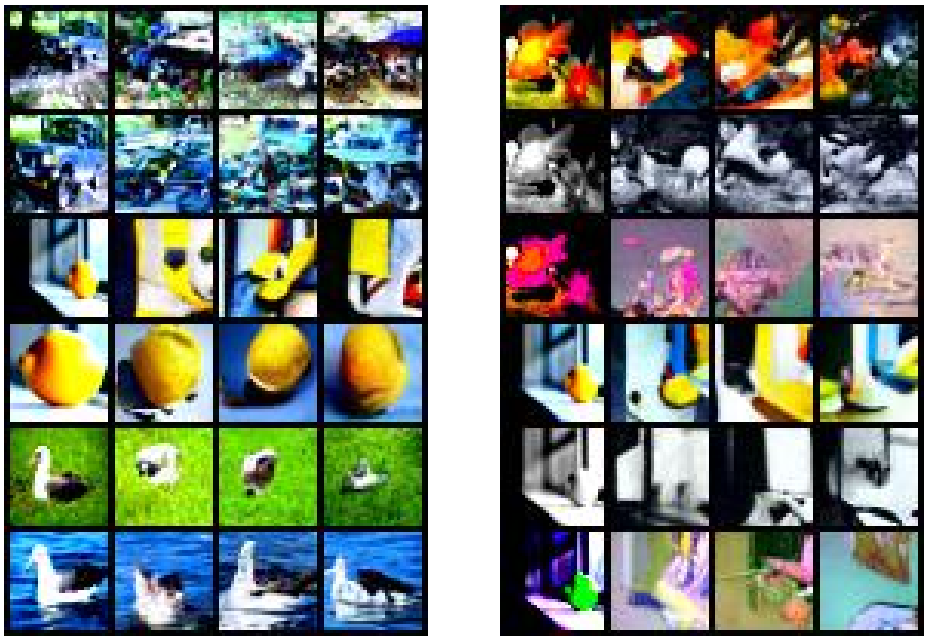

Figure 1: Data augmentation experiments using the **supervised model**. The first column shows the images given as input to the model to compute the representations. The second, third, and fourth columns show the corresponding samples generated by our RCDM model. **Left Image**: how the size of the object and the background affect the generated samples. **Right Image**: how greyscale and color jitter augmentations affect the generated samples.

samples decreases, however, we cannot observe the generated samples shifting from one class to another one.

### 6.4 FID score

The goal of this project is not to generate high-quality images but only to study the representation space of a model. For completeness however, we report the FID score of our RCDMs computed on 500 samples.

| Model | FID |
|---|---|
| Supervised | 282 |
| Dino (Backbone) | 261 |
| Dino (Head) | 252 |

These scores are very bad compared to the ones reported in the original paper. Of course, the quality of our samples is much worse. However, it is also worth to mention that the FID score depends on the number of samples used to compute it [8]. The original paper used 10k samples. We could not use such a high number of images since the time needed to compute the score would increase considerably (about 20 hours per model with our GPU).

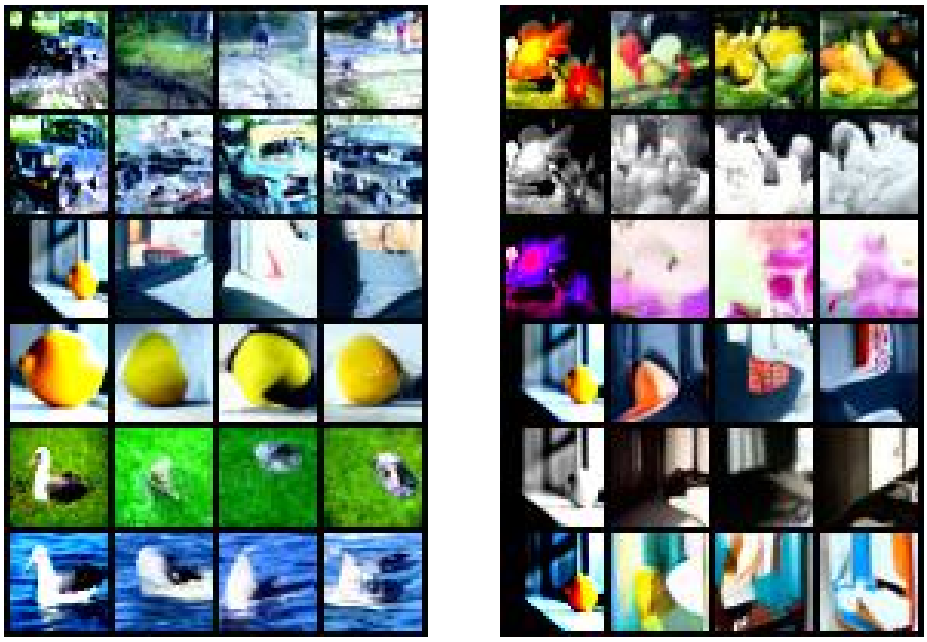

Figure 2: Data augmentation experiments using the **Dino model (backbone)**. The first column shows the images given as input to the model to compute the representations. The second, third, and fourth columns show the corresponding samples generated by our RCDM model. **Left Image**: how the size of the object and the background affect the generated samples. **Right Image**: how greyscale and color jitter augmentations affect the generated samples.

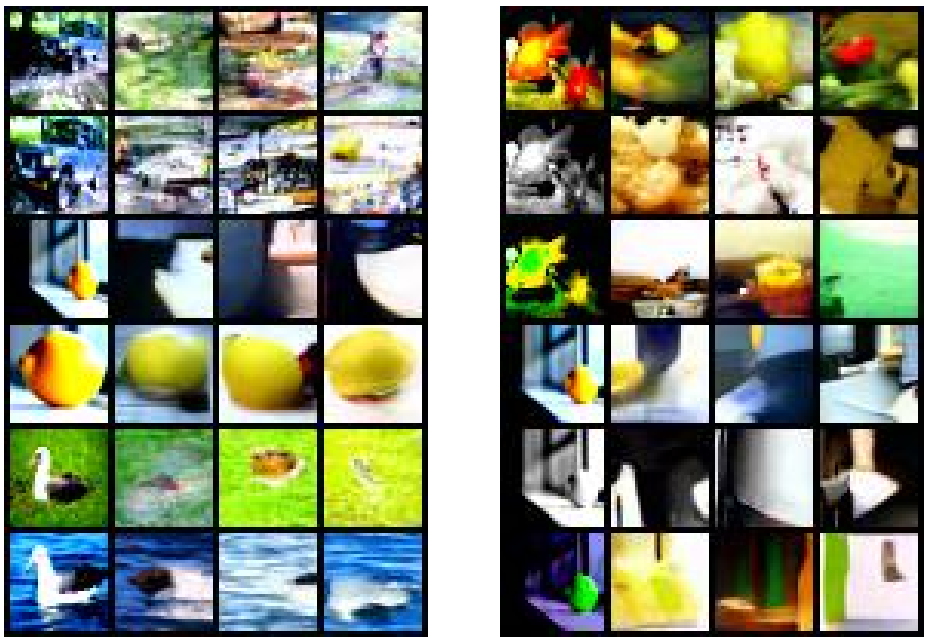

Figure 3: Data augmentation experiments using the **Dino model (Head)**. The first column shows the images given as input to the model to compute the representations. The second, third, and fourth columns show the corresponding samples generated by our RCDM model. **Left Image**: how the size of the object and the background affect the generated samples. **Right Image**: how greyscale and color jitter augmentations affect the generated samples.

Figure 4: Data augmentation experiments using the **Dino model (Backbone)** trained on a **different dataset** containing 50 similar classes (only animals). The first column shows the images given as input to the model to compute the representations. The second, third, and fourth columns show the corresponding samples generated by our RCDM model. **Left Image**: how the size of the object and the background affect the generated samples. **Right Image**: how greyscale and color jitter augmentations affect the generated samples.

Figure 5: Samples generated using as condition the interpolation of the representation of two different images. The original images are shown in the first and in last column (duck and flower). The columns in the middle show the generated samples as we move in the representation space from the duck to the flower. First row: supervised model. Second row: Dino model (backbone). Third row: Dino model (Head).

Figure 6: How FGSM adversarial attack affects the generated samples. The first two rows concern the Dino (Backbone) model; the first row shows the noisy images given as input to the model and the second row shows the corresponding generated samples. The last two rows concern the supervised model; the third row shows the noisy images given as input to the model and the fourth row shows the corresponding generated samples. The epsilon parameter of the FGSM attack increases from 0 to 1 as we move from left to right.

## 7 Challenges

During the project, we faced many challenges mainly due to our limited resources. The results of the original paper were obtained using an extremely large amount of computational power. The models that we used were very large and it was not easy to understand how they work and how to adapt them to our project. Also loading and using the models was a challenge. Trying to perform the forward step on the diffusion model with 512x512 images and batch size greater than 1 would need more than 15GiB GPU memory. Moreover, trying to train the model on the whole Imagenet dataset is basically impossible with our resources, and we needed to train multiple such models.

## 8 Conclusion

Despite the low resolution of our images and the small size of the dataset, we were able to obtain some interesting results. Despite the fact that the quality of our samples is not excellent it is good enough to draw some general conclusions about the structure of the representation space. We again want to mention that our generative models were not trained on augmented data. Despite this, the generative models were able to infer such augmentations from the representation space. We believe that this is the most significant part of the project. Our models are not good enough to make a detailed comparison between supervised and SSL representation space. However, we could observe that the head representation space seems to be invariant to some augmentations, which is one of the main results of the original paper.

## 9 Ethical consideration and societal impact

Artificial intelligence and deep learning are growing fields with the potential for large societal impact, it is therefore important to take an ethical approach to the research and development of such systems. Towards this goal, the European Commission have developed the Ethics Guidelines for Trustworthy AI [9]. It contains seven key principles that one should consider:

1. **Human agency and oversight:** This point states that AI systems should support human autonomy and decision-making. The RCDM model upholds this perfectly since it helps us understand what is learned by neural networks, allowing for humans to make informed decisions based on model output.

2. **Technical robustness and safety:** AI systems should reliably behave as intended. Allowing for visualization of representations makes it easier to ensure that a model has learned what it should.
3. **Privacy and Data governance:** The original paper and this project has used the ImageNet dataset, a public dataset. To the knowledge of us this dataset contains no biases, inaccuracies, or errors that could heavly affect the results in a negative manner.
4. **Transparency:** We have to the best of our ability explained the model architectures and the training process in a transparent manner. We have successfully reproduced these processes from the original paper, which is a testament to the transparency of their method. The source code has been made public towards this end.
5. **Diversity, non-discrimination and fairness:** As mentioned in the point regarding privacy and data governance, we are not aware of any discriminatory biases in the ImageNet dataset.
6. **Societal and environmental well-being:** It is our belief that the RCDM model can have a positive societal impact by making current and future AI models more transparent and understandable.
7. **Accountability:** The developers of AI systems should be responsible for their models. We believe this is achieved by reporting our results truthfully, and by making the source code publicly available, clearly showing who is behind the code and what has been done.

## 10 Self Assessment

We believe that this project was extremely hard to replicate with our resources. To proceed with the project we had to downsample the images and drastically decrease the size of the dataset, but this inevitably affected the results. The size and the quality of the dataset were the most important factors to replicate the results and we had to renounce them. Despite these changes, the time required to train the models slowed the project's progress drastically, it took days to train each of the models. We ran out of our google cloud free credits before finishing the project; this also was a big challenge considering that the models we used were too big to be loaded on our local machines and that the large size of Imagenet did not make it easy to move the project from a platform to another one. Despite these challenges, we were able to replicate the most important results and we have introduced interesting discussion points. We believe that we should get an A for this project.

## References


[1] F. Bordes, R. Balestriero, and P. Vincent, "High fidelity visualization of what your self-supervised representation knows about," *arXiv preprint arXiv:2112.09164*, 2021.

[2] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," 2020.

[3] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, 2021.

[4] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.

[5] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *International Conference on Machine Learning*, pp. 8162–8171, PMLR, 2021.

[6] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

[7] M. Koziarski and B. Cyganek, "Impact of low resolution on image recognition with deep neural networks: An experimental study," *International Journal of Applied Mathematics and Computer Science*, vol. 28, no. 4, 2018.

[8] M. J. Chong and D. Forsyth, "Effectively unbiased fid and inception score and where to find them," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6070–6079, 2020.

[9] E. Commission, C. Directorate-General for Communications Networks, and Technology, *Ethics guidelines for trustworthy AI*. Publications Office, 2019.