Federated object detection for defence and security applications using realistic unbalanced heterogeneous data distributions

Muriel van der Spek, Arthur van Rooijen, Lotte Nijskens, Martin van Leeuwen, Henri Bouma, and Hugo J. Kuijf

TNO Intelligent Imaging, Oude Waalsdorperweg 63, the Hague, the Netherlands

ABSTRACT

There is often sensitive data in defence and security applications, making it difficult for organizations to share such data. This limits the training of artificial intelligence techniques, which typically require large, diverse datasets. Federated learning offers a solution by enabling organizations to collaboratively train models without sharing private data. However, existing research on federated learning often focuses on simple computer vision tasks, such as classification on balanced datasets, and rarely addresses more complex tasks involving realistic, heterogeneous data distributions, also known as non-IID (non-independent and identically distributed) data. In this work, we demonstrate a federated learning framework applied to various object detection tasks relevant to defence and security. These tasks are evaluated under different types of non-IID conditions, including quantity skew, label skew, and feature skew. The object detection tasks include number and symbol detection on UNO card corners, single-frame person and vehicle detection from an air-to-ground perspective using the VisDrone dataset, and small moving object detection in challenging environments. Experimental results show that federated models consistently outperform separately trained models in both IID and non-IID settings. In experiments involving the three types of skew, federated performance decreases as the data becomes more non-IID. However, our results still demonstrate the added benefit of federated training compared to separately trained models. These findings highlight the viability of federated object detection in real-world defence and security scenarios involving heterogeneous data.

Keywords: Artificial intelligence, Data heterogeneity, EDF STORE, Feature skew, Federated learning, Label skew, Non-IID data, Object detection, Privacy-preserving AI, Quantity skew, Small object detection, Temporal-yolo, VisDrone

1. INTRODUCTION

Federated learning (FL) is a distributed training technique that allows organisations to collaboratively train a machine learning model without data sharing.[1–6] Since training data do not need to be shared among organisations, it is considered to be more secure than conventional training. This is especially important in the domains of defence and security, where data sometimes cannot be shared at all.[7–9]

Two main difficulties arise when using federated learning for real-world tasks:

1. Non-IID: in real-world tasks, data are often non-independent and identically distributed (non-IID) amongst the clients (participating organisations). Training with non-IID data can result in client drift, where local client models do not generalise well to the entire dataset and federated aggregation of local models is difficult.[10]

2. Object detection: the most common computer-vision task used in federated learning is image classification. There is research on object detection available, but the number of sources compared to image classification is lacking.[11] Additionally, existing federated learning techniques designed for image-classification tasks do not necessarily generalise to object-detection tasks.

Corresponding author: Muriel van der Spek, e-mail: muriel.vanderspek@tno.nl

M. van der Spek, A. van Rooijen, L. Nijskens, M. van Leeuwen, H. Bouma, H.J. Kuijf, Federated object detection for defence and security applications using realistic unbalanced heterogeneous data distributions, Proc. SPIE, vol. 13679, (2025). https://doi.org/10.1117/12.3069682

In this work, we investigate the impact of non-IID data on three object-detection tasks. Non-IID data will be decomposed into three separate issues, namely: (1) quantity skew, where each client has a different amount of training data available, (2) label skew, where each client has a different amount of classes available or some classes are completely missing in the training data, and (3) feature skew, where each client has different features in the training data, for example caused by different camera angles, weather conditions, or time of day.[10] Three different object-detection tasks are used to assess the effect of non-IID data on model performance, being the detection of numbers and symbols on UNO cards,[12] the detection of people and vehicles in the VisDrone dataset,[13] and the detection of drones in an in-house defence dataset.

2. RELATED WORK

This section covers related work, where non-IID data is explained first, followed by conventional strategies to account for non-IID data. Additionally, research on object detection with non-IID data is included to bridge the gap between methods made for classification and the applicability for object detection.

2.1 Non-IID data

In this work, non-IID data is considered to be so-called 'statistical heterogeneous', which concerns the difference in data distributions amongst the various private datasets at each client.[14] There are three key types of non-IID that will be studied in this work, being (1) quantity skew, (2) label skew, and (3) feature skew.[10] System heterogeneity, which concerns differences in computational resources between clients,[2] is out-of-scope for the current work.

These types of skew can be summarized as follows:[10,14,15]

1. **Quantity skew**: This type of skew originates purely from the size of the client's training data. All datasets contain all classes, and the label distribution in each client is still the same as the original distribution.
2. **Label skew**: This type of skew originates from the clients not having the same (number of) classes or class instances. In the most extreme situation, this can lead to some clients not having certain classes in their training data.
3. **Feature skew**: Feature skew is the result of each client having different features in their training dataset. This can originate from multiple sources. For example, in the MNIST dataset,[16] different people have different handwriting, creating different features for the same classes.[14]

The Dirichlet distribution[17] is often used to simulate non-IID data for quantity skew and label skew.[10,18] A higher Dirichlet coefficient means that the distribution will likely be more uniform. For values less than one, it is very likely that there is one client that has the majority of the total samples (quantity skew) or the majority of the samples of a class (label skew). A higher Dirichlet coefficient results in a higher possibility for a uniform data distribution, and when the Dirichlet coefficient approaches infinity, the distribution is guaranteed to be IID.[19] For research on non-IID data in federated learning, a Dirichlet coefficient between 0.1 and 1 is often used to demonstrate moderate-to-heavy skew.[20,21]

2.2 Aggregation strategies for IID and non-IID data

There are many aggregation strategies to account for different types of skewed data. The types of skew may include quantity, label and feature skew, but may also extend to system heterogeneous situations where the different hardware devices cause skew. Research presenting these strategies for example measure the effectiveness of their proposed strategies based on convergence speed, performance, security, or hardware requirements, where they demonstrate their strategy on a specific configuration to simulate skew. Each strategy may outperform other strategies in some predefined criteria, but there is none that outperforms others in all types of data skew. An extensive overview can be found in the survey of Ma et al.[14] and the survey of Gutierrez et al.[22]

The most conventional aggregation strategy is Federated Averaging (FedAvg).[23] This method takes the weighted average of all client's parameters, where the weighing factor is related to the fraction of the size of the

client's local dataset compared to the sum of the data available over all clients. This method is designed to work well in IID data scenarios, but when there is non-IID data, the performance obtained with FedAvg can drop.[24]

A recent literature study[22] analysed the frequency with which popular methods are used. FedProx[24] is by far the most applied strategy to account for the negative effects caused by heterogeneous data, and is referred to as the state-of-the-art for non-IID data.[25] This method is a variation of FedAvg, and accounts for changes between the global model and client models by adding a proximal term to the loss function. This proximal term is a measure for the difference between the global model and the client's local model. By incorporating this term, the client's model updates are forced in the direction of the global model. FedProx performs especially well when there are 'stragglers',[24] which is a form of system heterogeneity where some clients may be much slower than others due to differences in hardware specifications or communication bandwidth. The performance increase of FedProx is not visible in the case of statistical heterogeneity (based on the performance on MNIST with 0% stragglers[24]).

One promising strategy that covers statistical heterogeneity is FedBN,[25] which is a method designed to tackle feature skew. In FedBN, all "normal" model layers are communicated and averaged using FedAvg, while the batch normalisation layers are not shared and updated locally only. This has several advantages, including the addition of security on the communication link (batch normalisation statistics may contain sensitive information on the local data). It tackles feature skew, since simply averaging the batch normalisation statistics between clients with feature skewed data will not straightforwardly result in the most optimal batch normalisation. Since the batch normalisation statistics are only updated locally, the resulting global model will be different in each client, where each client created a model that is optimal for the situation that they can expect.

2.3 Federated object detection with data skew

Research on federated learning with non-IID data is often focussed on image classification, resulting in less knowledge on federated object detection. This can be seen in the survey of Shenaj et al.,[11] where they have an extensive overview of classification examples, but they only give five examples for object detection.

Mainly in the world of self-driving cars, non-IID federated object detection is an active research field. In these use cases, the non-IID data originates from feature skew that arises from using different cameras on the vehicle, or camera streams from different days. The study of Urmonov et al.[26] gives a summary of federated applications in intelligent vehicles. Several aggregated schemes have been applied for object detection, including FedAvg,[23] FedProx,[24] FedDyn,[27] FedBN[25] and SCAFFOLD.[28] For more information on these methods, see the survey of Ma et al.[14]

Federated learning using data from different cameras may not only contain feature skew, but also label skew, since different cameras can capture classes at different frequencies. Papers on autonomous vehicles do not always mention to what extent they witnessed or measured the label skew. One paper that does mention the label distribution is Jallepalli et al.[29] They perform federated object detection with cameras on cars (which results in feature skew), where they additionally document the class instances per client (where the label skew becomes visible). The level of skew appears moderate and they did not appear to need any algorithm to account for non-IID data. A similar thing is mentioned by Su et al.,[30] where they notice how class imbalance influenced their performance. When the class imbalance is too severe, their model breaks.

In summary, research on federated object detection lacks behind in image classification. The non-IID data are often a natural result from the camera setup, and not something that is explicitly measured.

3. METHOD

In this section, we describe the methodology to implement our custom federated learning framework (Section 3.1) for object detection (Section 3.2) on various types of non-IID data (Section 3.3).

(a) UNO (b) VisDrone (c) TNO-VOD

Figure 1: Three example images of the datasets used for (a) UNO, (b) VisDrone, and (c) TNO-VOD. The TNO-VOD example shows a zoomed region to illustrate how small the objects are.

3.1 Federated learning framework

The method implemented in this research revolves around a federated learning framework, where a single server coordinates multiple clients. These clients possess their local datasets, which may exhibit quantity skew, label skew, or feature skew. The implementation of the federated learning framework is an extension of prior work.[8]

The framework has two main components, the server and the clients. Each client trains its local model on its respective private dataset and sends updates to the server. These updates can be model weights or gradients. The server is responsible for aggregating local updates from clients and broadcasting the aggregated global model. Communication is managed by the tno.mpc.communication package.[31] Both is possible, but, we choose to communicate model weights rather than gradients. This gives the possibility to communicate both every single batch or every multiple batches or epochs, which will speed up the total training time. When gradients are communicated over multiple batches, additional logic is needed to combine information.

3.1.1 Aggregation

As already explained in the related work (Section 2.2), there is no non-IID aggregation strategy that outperforms the others in all specific situations. Aggregation is done by FedBN,[25] because it has three main advantages: (1) it can account for specifically feature skew, (2) it can be combined with FedAvg,[23] and (3) it adds security to the communication links by not communicating statistics on private data. Note that the comparison between FedAvg and FedBN on the three object-detection tasks on the three types of skew, respectively as explained in Section 3.2 and 3.3 is not in the scope of this research.

3.2 Federated object-detection tasks

This section covers the experimental setup for the three object-detection tasks. The first task is UNO, where the objective is to detect the numbers and symbols in the upper left corner of UNO cards.[12] Every image contains a stack of three UNO cards (Figure 1a). A Faster R-CNN with pre-trained ResNet-50 backbone is used in this task.[32] Note that a Faster R-CNN has frozen batch norm layers by default,[33] thus the FedBN implementation will be the same as plain FedAvg. The second task is VisDrone,[13] where the objective is to detect persons and vehicles from an air-to-ground perspective in single video frames (Figure 1b). A YOLOv8s[34] model was used in this task and fine-tuned on the training dataset split.[35] The third task is TNO-VOD (Video Object Detection), where the objective is to perform small moving object detection on videos (i.e. drone detection). Temporal-YOLOv8[36,37] was used in this task and trained on an in-house dataset covering both civilian and military tasks. This data was previously used for small (moving) object detection[38] (Figure 1c).

For both UNO and TNO-VOD, the federated learning setup contains two clients and one server. For VisDrone, the federated setup contains four clients and one server. For the UNO dataset, unless indicated otherwise, a subset of 62 images (showing a total of 186 UNO cards) was randomly selected, providing 31 images per client in the federated setup. The VisDrone training dataset contains 6,471 images, providing 1,617 images per client. The TNO-VOD dataset contains multiple videos, providing about 2,000 annotated drones per client.

Parameter synchronisation is done in different intervals for each task. UNO first trains on ten local epochs before synchronising the weights. In VisDrone and TNO-VOD, synchronisation is performed every batch. In VisDrone, there is a batch size of 10 images, and in TNO-VOD, the batch size is 30.

(a) Dominant class is "vehicle".

(b) Dominant class is "person".

Figure 2: Visualisation of two example VisDrone images used for the label skew experiments. The dominant class is determined (class that occurs most often in the image), and the other class is occluded by black boxes. Grey boxes are included to block out busy regions that lead to many detection mistakes, according to the original paper.[13]

3.3 Non-IID data

The federated learning setup as described in the previous subsection will be used to train three object detection tasks (UNO, VisDrone, TNO-VOD) in a federated way with non-IID data. Several types of skew are created to obtain non-IID data, namely quantity skew (Section 3.3.1), label skew (Section 3.3.2), and feature skew (Section 3.3.3). The next subsections will go into detail on how to obtain the skewed data distributions for each of the object-detection tasks, including the modifications that need to be made to the dataset (if needed).

3.3.1 Quantity skew

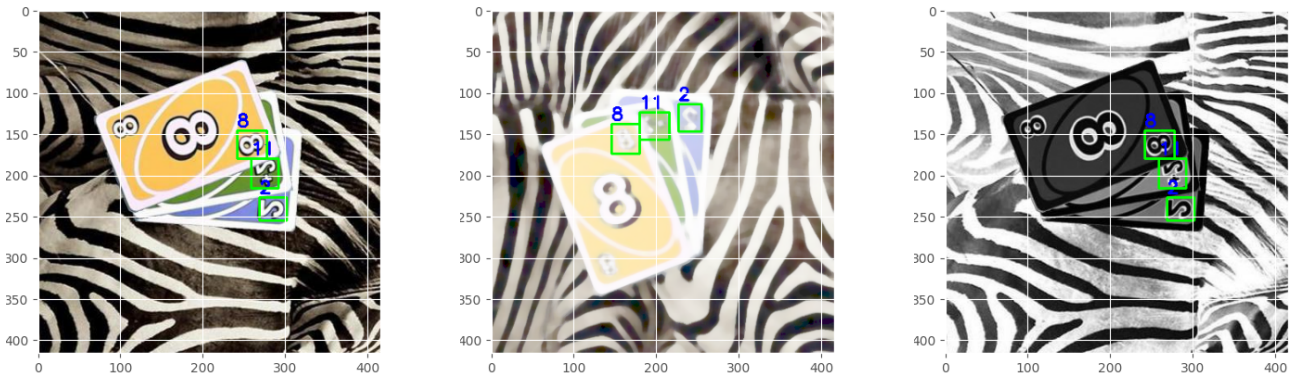
Quantity skew was created using a Dirichlet coefficient, where the general principle is similar for UNO and VisDrone. Here, the Dirichlet distribution defines how many samples are assigned to each client. Conventional values for the Dirichlet coefficient are in the range of 0.1 to 1.0, to demonstrate moderate/heavy skew. For TNO-VOD, the data contains videos, and quantity skew can be obtained by using videos of different lengths in the client's training dataset, where longer videos also contain more drone instances. For quantity skew, the distribution of the classes is the same for all client's dataset (and the original total dataset).

3.3.2 Label skew

Label skew was introduced in UNO and VisDrone. Not for TNO-VOD, since in this data there is only one class, making this type of skew not relevant. For UNO, the non-IID data is created by removing all images with a predefined label after assigning the data in an IID manner. The images themselves are not altered. In VisDrone, each image typically contains both persons and vehicles. To create label skew, instances of the less occurring class in each image are blocked out by drawing black boxes over them, and the corresponding annotations are removed. Two example images are shown in Figure 2. These modified images, containing only one class, were distributed to clients using a Dirichlet coefficient. A Dirichlet coefficient in the range of 0.1 to 1.0 can be used to show moderate-to-heavy skew.[20,21]

3.3.3 Feature skew

The last non-IID data type covered is feature skew and can be introduced in multiple ways. In UNO and VisDrone, feature skew is added by adding augmentations to the client's training data. In UNO, this is done by using colour inversion and conversion to greyscale in the first client, and light exposure and blurring in the second client (Figure 3). Feature skew is created in VisDrone by adding noise to the training data of the clients, where each client has a different amount of noise (Figure 4). For TNO-VOD, testing feature skew is very ambiguous. One possible type of feature skew is letting clients have videos from different cameras, referred to as real-world feature skew. However, if the data are similar, the actual feature skew in the data is not very measurable. Therefore, it is chosen to not explicitly test feature skew for TNO-VOD.



(a) Basic augmentation. (b) Overexposure and blurring. (c) Colour inversion and greyscale.

Figure 3: Three example types of augmentations. (a) shows "standard" augmentation, including rotation and light blurring. The other two images additionally include overexposure and heavy blurring (b), and (c) colour inversion and greyscale. The last two can be used to add feature skew in the client's datasets.

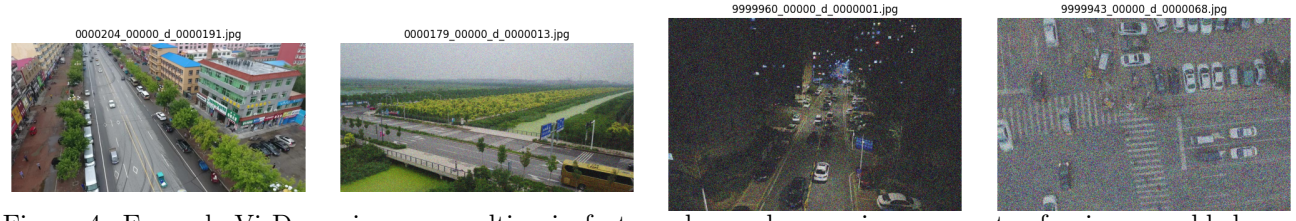


Figure 4: Example VisDrone images resulting in feature skew, where various amounts of noise are added over the images. The severity of the noise increases from left to right, where the first image does not contain any noise.

4. RESULTS

This section presents the results of UNO, VisDrone, and TNO-VOD on the three types of skew. First, baseline performances are established on IID data, followed by the results on quantity skew, label skew, and feature skew.

4.1 Baselines on IID data

To assess the added value of federated learning in three object-detection tasks, we first establish a performance baseline. This involves comparing training on the full dataset (ideal scenario) versus training on partial data subsets (worst-case scenario, where organizations cannot share data). This comparison is crucial to demonstrate the impact of federated learning. If both full and separate datasets yield already similar performance, there is little added benefit of access to more data that a federated learning approach would offer.

The goal is to identify the point where performance drops significantly with reduced data, highlighting where federated learning becomes valuable. This is done through three experiments:

- Centralised training: Training on the full dataset, representing the upper performance bound.
- Federated training: Training across two (UNO, TNO-VOD) or four (VisDrome) clients with balanced (IID) data splits, synchronising weights every ten epochs (UNO) or every batch (VisDrone, TNO-VOD).
- Separate training: Training on 50% (UNO, TNO-VOD) or 25% (VisDrone) of the dataset without collaboration, representing the lower performance bound.

In the remainder of this paper, when we refer to centralized/federated/separate results or models, we mean models that are trained using the descriptions above.

Table 1: Baseline object-detection results on different datasets (UNO, VisDrone and TNO-VOD). The centralized performance (on all training data), separate performance (on a subset), and federated performance (federated learning with sharing between clients) are compared. Note that for UNO mAP@[.5:.95] is used, where VisDrone and TNO-VOD use mAP@.5. This is because of the differences in complexity of the tasks, leading to different relevancy for the metrics.

Configuration	mAP@[.5:.95] UNO	mAP@.5 VisDrone	mAP@.5 TNO-VOD
Centralized	0.70	0.66	0.65
Federated	0.64	0.66	0.65
Separated	0.50	0.63	0.60

We expect centralised training to perform best, separate training worst, and federated training to fall in between (but closer to centralised). Results for UNO, VisDrone, and TNO-VOD are shown in Table 1. For UNO, the mAP@[.5:.95] metric is used, which for this relatively easy task gives a reliable indication of the detection performance For VisDrone and TNO-VOD mAP@.5 is used, because in these tasks the objects are very small, making mAP@[.5:.95] scores relatively low whilst detection performance is generally sufficient.

From this table, it becomes visible that for all three object-detection tasks, federated learning has an added benefit. In all three experiments, the federated performance is higher than the separate performance, and even close to/the same as the centralized performance.

For UNO, the gap between separated and centralized training is the biggest, which is a result of using a small partition of the full UNO-Cards dataset to be able to show the effect of federated learning (if the full dataset was used, all three performances would be at a maximum). The federated UNO result is the only result that is not the same as the respective centralized result of each experiments, which may be caused by the dataset being very small. However, since the performance is still close to the centralized performance, it is safe to conclude that for UNO federated learning is also beneficial.

Note that in the table, our mAP@.5 for both single-frame object detection and small object detection seem low compared to the UNO mAP@[.5:.95] score, given that UNO here is only trained on 63 images. VisDrone and TNO-VOD are more difficult than UNO and the objects to detect are much smaller, given that it becomes harder to have overlap between the target and predicted bounding boxes. mAP@.5 results from literature for the centralized performance on VisDrone with YOLOv8s is approximately 0.57.[39] Our result is already higher, namely 0.66 compared to 0.57, making it safe to continue with the non-IID experiments. One explanation for the difference could be that they did not block out the busy areas (grey boxes in Figure 1b) that can lead to many detection errors. For TNO-VOD, there is no reference from literature, since in-house data is used, but we validate our results since the performance on TNO-VOD is similar to VisDrone.

4.2 Quantity skew

The next experiments are conducted to show the effect of quantity skew in each of the three datasets: UNO (Section 4.2.1), VisDrone (Section 4.2.2) and TNO-VOD (Section 4.2.3).

4.2.1 UNO

This section demonstrates the results UNO with amounts of quantity skew in the training data. The hypothesis is that separate clients, each with limited data, will experience a drop in performance when trained separately (under the assumption that the local dataset is small). In contrast, the performance of the federated model is expected to remain relatively stable compared to the IID baseline. The results are shown in Table 2, where the first two performance columns represent the local performance in the clients (separate - no parameter synchronisation), and the last column is the federated performance when there is parameter synchronisation.

Two effects are visible from this table: (1) the separate performances of the clients decrease when the dataset becomes smaller, and (2) the federated performance stays rather constant and unaffected by the variations of quantity skew present. The federated performance even appears to increase slightly towards the centralized upper bound of 0.70 from Table 1 as the data becomes more unbalanced. This may be due to the properties

Table 2: Performance of federated object detection on UNO cards with quantity skew. Separate mAP refers to local client performance on a subset of the training data, and federated mAP reflects performance after federated training.

Data distribution	Samples		Separate mAP@[.5:.95]		Federated mAP@[.5:.95]	
	Client 0	Client 1	Client 0	Client 1	Client 0	Client 1
IID	31	31	0.50	0.50	0.64	0.64
Dirichlet(0.5)	37	26	0.49	0.47	0.64	0.64
Dirichlet(0.3)	50	13	0.61	0.33	0.65	0.65
Dirichlet(0.1)	62	1	0.70	0.00	0.67	0.67

Table 3: Data distribution and resulting performance for VisDrone across four clients under various quantity skew settings. Separate mAP refers to local client performance on a subset of the training data, and federated mAP reflects performance after federated training. The standard deviation is obtained with the results of the four clients over one run.

Data Distribution	Number of images per client				Separate mAP@.5	Federated mAP@.5
	Client 0	Client 1	Client 2	Client 3		
IID	1617	1617	1617	1617	0.63 ± 0.01	0.66 ± 0.00
Dirichlet(0.5)	4501	1486	470	12	0.48 ± 0.30	0.52 ± 0.25
Dirichlet(0.3)	5774	625	39	33	0.39 ± 0.28	0.45 ± 0.20
Dirichlet(0.1)	4184	2287	3	1	0.33 ± 0.37	0.37 ± 0.30

of FedAvg, which includes a weighting factor based on the size of each client's local dataset. When the data is IID, both clients contribute equally (each with 50% weight). As the data split becomes more unbalanced, the parameters from the client with the larger dataset carry more weight than those from the client with less data. In the baseline experiments, centralized training outperformed federated training. Therefore, when one client has a dataset size approaching that of the centralized setup, while the other client's data is significantly reduced, it is expected that the global performance will approach the centralized baseline of 0.70.

4.2.2 VisDrone

This subsection presents the results of VisDrone with various amounts of quantity skew in the training data. This experiment is very similar to UNO, but the main difference is that the VisDrone data has more (complex) features. We hypothesise that the individual federated clients will not be able to generalise to the global objective if their training datasets are too small, since they will not be able to establish reliable batch normalisation statistics. The federated performance is the average of the performance on the four clients global models, and if there is one client that did not see sufficient data, the average federated performance will be affected. The results of the experiments with quantity skew are shown in Table 3.

Generally, in this table it can be seen that all performance metrics decrease when the data becomes more non-IID, which is in line with the expectations. However, this is different than the effect that was visible in the UNO experiment, where the performance increased when the data becomes more non-IID. This may be explained by the fact that for VisDrone, the federated and centralized performance are already the same, thus a further performance increase in a more non-IID setting would not be visible. Additionally, looking at the performance of individual clients shows that clients with a lot of images in their training data are able to obtain the maximum performance, while clients with not enough data break and lower the average that is presented in Table 3. The performances of the individual clients are presented in Appendix A. Inspecting the individual federated client's performances, two phenomena are visible: (1) clients with less than 40 samples are not able to generalize, and (2) clients with sufficient data achieve the same performance as the IID result (or the centralized baseline results). We expect that this happens because 40 images or less is not enough to create reliable batch normalisation parameters, breaking these models. This could be a downside of FedBN, because sharing and averaging the batch normalisation statistics is expected to solve this issue here, which would be a good evaluation for future

Table 4: Data distribution and corresponding federated performance (mAP@.5) for TNO-VOD across two clients under quantity skew settings. Separate mAP refers to local client performance on a subset of the training data, and federated mAP reflects performance after federated training. The number of instances in the video's per client is shown.

Data Distribution	Samples per client		Separate mAP@.5		Federated mAP@.5	
	Client 0	Client 1	Client 0	Client 1	Client 0	Client 1
Quantity skew	2,500	20,000	0.39	0.61	0.51	0.59

work. Interestingly, clients with enough data are not broken by these clients that are not able to learn, implying that the setup is able to handle quantity skew when there is enough data present.

4.2.3 TNO-VOD

This subsection discusses the quantity skew results of TNO-VOD, where the length of the videos assigned to clients is used to obtain skewed data. The expectation is that the individual performance of the client with the shorter video performs worse than the client with a longer video, since it saw less samples. The federated performance is expected to be better than the separate performance. The results are shown in Table 4. Note that these results can not be compared with the baseline results from Table 1 because different videos are used in the clients.

From this table, two things are visible. First, the longer video gives better performance than the shorter video, which is a logical result of seeing less samples. Second thing is that federated client 0 is boosted compared to separate client 0, where client 1 loses 0.02. Whether this small drop is statistically significant is not explored, but generally it is clear that federated learning has added value because of the performance increase of client 0, and only a small performance reduction in client 1.

4.2.4 Summary on quantity skew

Experimental results with quantity skew proved to be interesting. A general trend among all object-detection tasks is visible, where the separate performance is lower than the federated performance, showing the added benefit of federated learning. In UNO, the federated performance was stable, regardless of the data distribution. In VisDrone, the federated performance drops with the amount of data in the clients. The separate performance of the clients is still lower than the federated performances, showing the added benefit of federated learning even in situations where some clients have very small datasets. One effect that is visible however is that clients with too little data samples are not able to obtain reliable batch normalisation statistics, creating a federated model that is broken. Clients with enough data are able to obtain performance similar to the IID performance, meaning that these models are not broken by clients that are not able to learn. In TNO-VOD, only a small performance decrease for the client with a large dataset is visible, where the client with little data shows a big improvement. All these results imply that federated object detection with FedBN aggregation is able to handle quantity skew.

4.3 Label skew

This section covers the results for UNO (Sec. 4.3.1) and VisDrone (Sec. 4.3.2) when the data contains label skew.

4.3.1 UNO

This subsection covers the results for object detection on the UNO cards when the data between the clients contain label skew. The aim is to show the effect on the global performance when classes are missing in one client. Other experiments used 1% of the UNO-Cards dataset to demonstrate FL capabilities, but this experiment uses 5%, because otherwise too few samples per class are left in classes that are missing in one of the clients. The class distribution per client is shown in Figure 5, where the training data of the first client does not contain class 2 and 3, and the second client does not have 9 and 10. The results for this experiment are shown in Table 5.

The results in Table 5 show that the federated model is capable of predicting the bounding boxes of all classes, even if one of the clients was missing data. On average over all classes, the performance of the federated approach is higher than the separate performances. The only exception is class 10, where the federated model

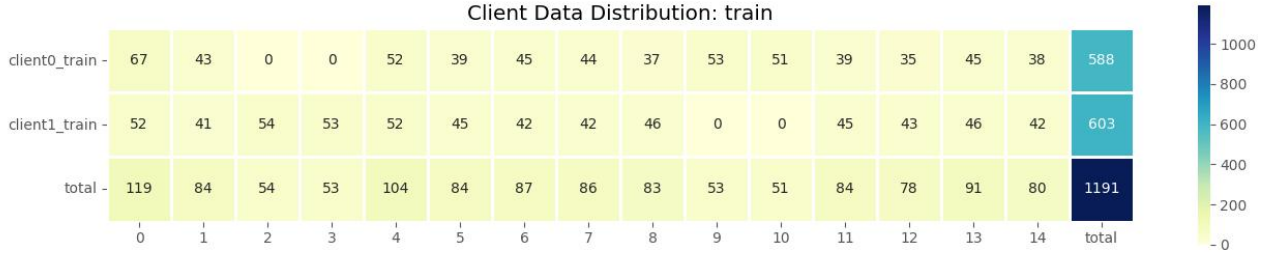


Figure 5: Number of samples of training data for each client. One client lacks train data for classes 2 and 3, and the other for classes 9 and 10.

Table 5: Performance of UNO with label skew. Separate mAP refers to local client performance on a subset of the training data, and federated mAP reflects performance after federated training. The performance is documented on classes with training data in one of the clients, classes with training data in both clients, and an average performance on all classes.

Class IDs	Notes	Separated mAP@[.5:.95]		Federated mAP@[.5:.95]
		Client 0	Client 1	
2	Class only in first client	0.00	0.80	0.80
3	Class only in first client	0.00	0.53	0.61
9	Class only in second client	0.78	0.00	0.79
10	Class only in second client	0.72	0.00	0.63
0-1, 4-8, 11-14	All common classes	0.73	0.63	0.79
0-14	All classes	0.63	0.50	0.77

performs worse than client 0. Inspecting the confusion matrix (not shown here) shows that client 0 in almost one out of three predictions for class 10, it actually predicts class 3, explaining why client 0 performs better on class 10 without using federated learning.

4.3.2 VisDrone

This subsection covers the results of label skew in VisDrone. Performance is expected to drop as the amount of heterogeneity increases. This is similar to the results for VisDrone with quantity skew, where it became clear that clients with too little data are not able to obtain reliable batch norm statistics. Additionally we expect the separate performance to be lower than the federated performance In the label skew situation, this may be even more prominent, since there is only one class per image, reducing the amount of information in the images. The results are shown in Table 6.

The results show that for almost all settings, the federated models perform better than the separate models, showing the added benefit of federated learning in when there is label skew. The performance of the individual clients is documented in Appendix B, where this effect is also visible on a client level. Created skew with Dirichlet(0.5) is the same as the baseline, and the federated performance when applying skew with Dirichlet(0.3) and

Table 6: Results for VisDrone with label skew. Separate mAP refers to local client performance on a subset of the training data, and federated mAP reflects performance after federated training. The standard deviation is obtained with the results of the four clients over one run.

Data Distribution	Separated mAP@.5	Federated mAP@.5
IID	0.53 ± 0.01	0.56 ± 0.01
Dirichlet(0.5)	0.50 ± 0.04	0.56 ± 0.02
Dirichlet(0.3)	0.47 ± 0.03	0.53 ± 0.01
Dirichlet(0.1)	0.44 ± 0.13	0.51 ± 0.04

Table 7: Performance of object detection on UNO cards with feature skew created with data augmentations. Separate mAP refers to local client performance on a subset of the training data, and federated mAP reflects performance after federated training.

Data distribution	Separated mAP@[.5:.95]		Federated mAP@[.5:.95]
	Client 0	Client 1	
IID	0.50	0.50	0.64
Feature skew	0.36	0.35	0.54

Dirichlet(0.1) did not appear to drop in performance a lot. Generally, our hypothesis was that the performance decreases when the data becomes more unbalanced. However, the drop is less than we expected. This could either be that the setup indeed is very robust against label skew, or maybe more likely, the experiment did not show the most extreme label skew. When there is extreme label skew in the conducted experiments, there are is one or two clients that posses all data, where the other clients would almost not participate in federated training due to the weighing factor of FedAvg (which is included in FedBN). To test this, it would be interesting to conduct an experiment where there are more classes, and maybe also more clients, where each client has a different distribution of these classes. Then, the skew is caused mainly by label imbalance, and is less overruled by quantity skew.

4.3.3 Summary on label skew

Label skew experiments reveal that federated learning can effectively generalize across unbalanced class distributions using two example object-detection task. In UNO, the federated model is still able to predict classes correctly, even though the classes were only present in one of the clients. For VisDrone, federated models generally outperformed separate models. Overall, the results confirm that federated learning remains effective when there is label skew.

One interesting future direction would be to include experiments with more classes in VisDrone, because now the problem may arguably come close to the quantity skew experiments where one or two clients have all data, and the others do not participate much (which is a property of FedAvg, which is included in FedBN). This effect would be less prominent when multiple clients would have different number of samples of may clients, where each client contributes a similar amount to the global federated model.

4.4 Feature skew

This section covers the results of federated learning with feature skew between the client's datasets on UNO (Section 4.4.1) and VisDrone (Section 4.4.2).

4.4.1 UNO

Feature skew on the UNO dataset is obtained by using different augmentations on the private dataset of a client (as was shown in Figure 3). The hypothesis is that the resulting federated performance on the skewed data is a bit lower than the IID data, because it is expected that training is made more difficult. The results are shown in Table 7.

From this table, we see that adding the augmentations makes training more difficult, because the feature skew results are lower than the IID results. Nevertheless, the federated model outperforms the individual (client 0 and 1) models. Also, in the IID setting, the federated performance is better than the separate performance. Therefore, it is safe to conclude that for UNO card detection, federated learning can account for feature skew, and the performance is mainly affected by the training task being more difficult.

Table 8: Performance metrics for VisDrone of individual clients under noise-based feature skew. Separate mAP refers to local client performance on a subset of the training data, and federated mAP reflects performance after federated training.

Skew type	Client index	Separated mAP@.5	Federated mAP@.5
IID	Average	0.63	0.66
Feature skew	Client 0	0.64	0.64
	Client 1	0.58	0.62
	Client 2	0.49	0.58
	Client 3	0.41	0.53
	Average	0.53 ± 0.09	0.59 ± 0.04

4.4.2 VisDrone

This subsection shows the results for noise-based feature skew on VisDrone (as was shown in Figure 4). Here the hypothesis is similar to the UNO feature skew experiment. We expect training to be more difficult, resulting in lower performance for federated feature skew than the IID results, because the noise will make it more difficult to detect persons and vehicles. The results are shown in Table 8, where the separate results (without parameter synchronisation) and federated results can be compared From this table it is visible that for all clients, the federated performance is higher than (or equal to) the separate result, which is in line with our expectations. Additionally, the separate performance drops with the amount of noise, but training federated seems to give these clients a boost, even though the batch normalisation parameters are not shared. This is very interesting and it is expected that the usage of FedBN makes sure that the clients with a lot of noise dont break the global model. For future research, it would be interesting to add a comparison study of the effect of FedAvg versus FedBN.

4.4.3 Summary on feature skew

Feature skew experiments on both UNO and VisDrone demonstrate that federated learning remains effective even when clients have different features in their training data. In UNO, applying different augmentations per client led to a drop in performance compared to the IID setting, showing that feature skew makes training more difficult. Fortunately, the federated model still outperformed the separate models, proving that the different augmentations do not break the global model, where they did arguably break the separately trained models. Similarly, in VisDrone, feature skew caused a decline in separate model performance, especially for clients with more noise. Here, the federated performance is again better than the separate models. We expect that this is because of the FedBN being able to handle differences in local feature representations. These findings suggest that while feature skew introduces challenges, federated learning can mitigate its impact and produce stronger global models than separate training.

5. CONCLUSION

This study demonstrates federated learning for object detection in defense and security applications, particularly under non-independent and identically distributed (non-IID) data distributions. By evaluating three representative tasks, namely UNO card detection (UNO), single-frame object detection (VisDrone), and small moving object detection (TNO-VOD), the impact of quantity skew, label skew, and feature skew is analysed on the federated performance.

Our results show that federated learning consistently outperforms separate training across all tasks and skew types, and in many cases approaches the performance of IID training. While quantity skew had minimal impact on simpler tasks like UNO, it resulted in a performance drop in VisDrone and TNO-VOD, especially when clients had insufficient data to establish reliable batch-normalization statistics. Label-skew experiments revealed that the federated UNO model could generalize to unseen classes, and experiments on VisDrone also showed that the federated model still learns all classes, even though the classes are non-IID. Feature skew, particularly when induced through augmentations or noise, led to divergent client models due to local batch normalization, but

the federated model still outperformed individual clients. The federated model in both UNO and VisDrone is not broken by the feature skew, while individual clients perform a lot worse.

These results give interesting insights on the behaviour of multiple types of object-detection tasks across various types of non-IID data using FedBN. Future work will explore more advanced aggregation strategies to account for performance drops caused by non-IID data. Additionally, it would be interesting to explore the range of behaviour further by running the experiments more often, to reliably estimate significance and try more Dirichlet coefficients to analyse the pattern more precisely. The latter is especially interesting for label skew in VisDrone, which was the experiment for which the most unexpected behaviour was present. For future work, we plan to investigate the comparison of FedBN with FedAvg and other aggregation strategies, to explore the behaviour of these methods on non-IID object detection even further.

ACKNOWLEDGMENTS

This work was performed in the projects EDF-STORE and ERP Next-Generation Crypto. This work received funding from the the European Defence Fund through the project STORE (Shared daTabase for Optronics image Recognition and Evaluation), grant agreement № 101121405.

REFERENCES

[1] J. Liu, J. Huang, Y. Zhou, X. Li, S. Ji, H. Xiong, and D. Dou, "From distributed machine learning to federated learning: A survey," *Knowledge and information systems* **64**(4), pp. 885–917, 2022.

[2] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowledge-Based Systems* **216**, p. 106775, 2021.

[3] L. Li, Y. Fan, M. Tse, and K.-Y. Lin, "A review of applications in federated learning," *Computers & Industrial Engineering* **149**, p. 106854, 2020.

[4] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE signal processing magazine* **37**(3), pp. 50–60, 2020.

[5] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečnỳ, S. Mazzocchi, B. McMahan, *et al.*, "Towards federated learning at scale: System design," *Proceedings of machine learning and systems* **1**, pp. 374–388, 2019.

[6] P. Kairouz, H. B. McMahan, B. Avent, *et al.*, "Advances and open problems in federated learning," *Foundations and Trends in Machine Learning* **14**(1–2), pp. 1–210, 2021.

[7] K. Demertzis, P. Kikiras, C. Skianis, K. Rantos, L. Iliadis, and G. Stamoulis, "Federated auto-meta-ensemble learning framework for ai-enabled military operations," *Electronics* **12**(2), p. 430, 2023.

[8] M. van der Spek, A. van Rooijen, and H. Bouma, "Secure sparse gradient aggregation with various computer-vision techniques for cross-border document authentication and other security applications," in *Artificial Intelligence for Security and Defence Applications II*, **13206**, pp. 121–134, SPIE, 2024.

[9] S. B. van Rooij, M. van der Spek, A. van Rooijen, and H. Bouma, "Privacy-preserving federated learning with various computer-vision tasks for security applications," in *Artificial Intelligence for Security and Defence Applications*, **12742**, pp. 23–35, SPIE, 2023.

[10] Q. Li, Y. Diao, Q. Chen, and B. He, "Federated learning on non-iid data silos: An experimental study," in *2022 IEEE 38th international conference on data engineering (ICDE)*, pp. 965–978, IEEE, 2022.

[11] D. Shenaj, G. Rizzoli, and P. Zanuttigh, "Federated learning in computer vision," *Ieee Access* **11**, pp. 94863–94884, 2023.

[12] Roboflow, "Uno cards dataset." https://public.roboflow.com/object-detection/uno-cards, 2020. Shared by Adam Crawshaw, Accessed: 2025-07-07.

[13] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, and H. Ling, "Detection and tracking meet drones challenge," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **44**(11), pp. 7380–7399, 2021.

[14] X. Ma, J. Zhu, Z. Lin, S. Chen, and Y. Qin, "A state-of-the-art survey on solving non-iid data in federated learning," *Future Generation Computer Systems* **135**, pp. 244–258, 2022.

[15] H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-iid data: A survey," *Neurocomputing* **465**, pp. 371–390, 2021.

[16] L. Deng, "The mnist database of handwritten digit images for machine learning research," *IEEE Signal Processing Magazine* **29**(6), pp. 141–142, 2012.

[17] K. W. Ng, G.-L. Tian, and M.-L. Tang, *Dirichlet and related distributions: Theory, methods and applications*, John Wiley & Sons, 2011.

[18] J. Zhang, Z. Li, B. Li, J. Xu, S. Wu, S. Ding, and C. Wu, "Federated learning with label distribution skew via logits calibration," in *International Conference on Machine Learning*, pp. 26311–26329, PMLR, 2022.

[19] H. Li, *Latent Dirichlet Allocation*, pp. 439–471. Springer Nature Singapore, Singapore, 2024.

[20] L. Gao, H. Fu, L. Li, Y. Chen, M. Xu, and C.-Z. Xu, "Feddc: Federated learning with non-iid data via local drift decoupling and correction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10112–10121, 2022.

[21] M. Luo, F. Chen, D. Hu, Y. Zhang, J. Liang, and J. Feng, "No fear of heterogeneity: Classifier calibration for federated learning with non-iid data," *Advances in Neural Information Processing Systems* **34**, pp. 5972–5984, 2021.

[22] D. M. J. Gutierrez, D. Solans, M. A. Heikkilä, A. Vitaletti, N. Kourtellis, A. Anagnostopoulos, and I. Chatzigiannakis, "Non-iid data in federated learning: A survey with taxonomy, metrics, methods, frameworks and future directions," *CoRR* **abs/2411.12377**, 2024.

[23] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*, pp. 1273–1282, PMLR, 2017.

[24] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems* **2**, pp. 429–450, 2020.

[25] X. Li, M. Jiang, X. Zhang, M. Kamp, and Q. Dou, "Fed{bn}: Federated learning on non-{iid} features via local batch normalization," in *International Conference on Learning Representations*, 2021.

[26] O. Urmonov, S. Sajid, Z. Aziz, and H. Kim, "Federated object detection scenarios for intelligent vehicles: Review, case studies, experiments and discussions," *IEEE Transactions on Intelligent Vehicles* , 2024.

[27] D. A. E. Acar, Y. Zhao, R. M. Navarro, M. Mattina, P. N. Whatmough, and V. Saligrama, "Federated learning based on dynamic regularization," in *International Conference on Learning Representations*, 2021.

[28] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *International conference on machine learning*, pp. 5132–5143, PMLR, 2020.

[29] D. Jallepalli, N. C. Ravikumar, P. V. Badarinath, S. Uchil, and M. A. Suresh, "Federated learning for object detection in autonomous vehicles," in *2021 IEEE Seventh International Conference on Big Data Computing Service and Applications (BigDataService)*, pp. 107–114, IEEE, 2021.

[30] S. Su, B. Li, C. Zhang, M. Yang, and X. Xue, "Cross-domain federated object detection," in *2023 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1469–1474, IEEE, 2023.

[31] T. P. Lab, "tno.mpc.communication: Secure multi-party computation (mpc) - communication." https://github.com/TNO-MPC/communication, 2025. Accessed: 2025-07-07.

[32] M. van der Spek, A. van Rooijen, and H. Bouma, "Secure sparse gradient aggregation with various computer-vision techniques for cross-border document authentication and other security applications," in *Artificial Intelligence for Security and Defence Applications II*, **13206**, p. 132060D, SPIE, 2024.

[33] T. Contributors, "Faster r-cnn — torchvision source code." https://docs.pytorch.org/vision/stable/_modules/torchvision/models/detection/faster_rcnn.html, 2023. Accessed: 2025-08-18.

[34] G. Jocher, J. Qiu, and A. Chaurasia, "Ultralytics YOLO," Jan. 2023.

[35] A. M. Liezenga, H. R. Baumann, S. Becker, S. Bensberg, H. R. Eiring, A. W. Johnsgaard, Z. R. Lee, M. Marturini, L. Nijskens, M. Rapp, J. E. van Woerden, A. Wolpert, T. Liiv, and H. J. Kuijf, "Comparative Study of Out-of-the-Box Technology for Automatic Target Detection and Recognition," in *Artificial Intelligence for Security and Defence Applications III*, 2025.

[36] M. C. van Leeuwen, E. P. Fokkinga, W. Huizinga, J. Baan, and F. G. Heslinga, "Toward versatile small object detection with temporal-yolov8," *Sensors* **24**(22), 2024.

[37] F. G. Heslinga, F. Ruis, L. Ballan, M. C. van Leeuwen, B. Masini, J. E. van Woerden, R. J. den Hollander, M. Berndsen, J. Baan, J. Dijk, *et al.*, "Leveraging temporal context in deep learning methodology for small object detection," in *Artificial Intelligence for Security and Defence Applications*, **12742**, pp. 134–145, SPIE, 2023.

[38] M. Van Lier, M. Van Leeuwen, B. Van Manen, L. Kampmeijer, and N. Boehrer, "Evaluation of spatio-temporal small object detection in real-world adverse weather conditions," in *Proceedings of the Winter Conference on Applications of Computer Vision*, pp. 844–855, 2025.

[39] H. Lou, X. Duan, J. Guo, H. Liu, J. Gu, L. Bi, and H. Chen, "Dc-yolov8: Small-size object detection algorithm based on camera sensor," *Electronics* **12**(10), p. 2323, 2023.

Table 9: Performance metrics for VisDrone of individual clients under different Dirichlet quantity skew settings. Separate mAP refers to local client performance on a subset of the training data, and federated mAP reflects performance after federated training.

Dirichlet α	Client index	Number of images	Separated mAP@.5	Federated mAP@.5
Dirichlet(0.5)	Client 0	4502	0.65	0.66
	Client 1	1486	0.64	0.66
	Client 2	470	0.59	0.65
	Client 3	12	0.03	0.08
	Average		0.48 ± 0.30	0.52 ± 0.25
Dirichlet(0.3)	Client 0	5774	0.65	0.66
	Client 1	625	0.61	0.65
	Client 2	39	0.06	0.25
	Client 3	33	0.26	0.25
	Average		0.39 ± 0.28	0.45 ± 0.20
Dirichlet(0.1)	Client 0	4180	0.65	0.67
	Client 1	2284	0.64	0.67
	Client 2	3	0.02	0.06
	Client 3	1	0.00	0.09
	Average		0.33 ± 0.37	0.37 ± 0.30

APPENDIX A. QUANTITY SKEW VISDRONE

Table 9 shows results for VisDrone and quantity skew. Each individual client in each Dirichlet setting is documented individually to highlight the difference between separate training and federated training. From the table, it becomes visible that for the separately trained clients, the performance drops when the amount of data becomes less. This effect is also visible in the federated clients. However, the performance is always higher than in the separate setting, implying that synchronising the weights is less affected by clients that do not have enough data. The federate clients that have sufficient number of images in their training data seem uplifted towards the IID baseline of 0.66, even though the clients individually were not able to achieve this performance when trained separately.

APPENDIX B. LABEL SKEW VISDRONE

Table 10 shows results for VisDrone with label skew. Each individual client in each Dirichlet setting is documented individually to highlight the difference between separate training and federated training. For all experiments (except for client 1 in Dirichlet(0.1)), the separate performance of the individual clients is lower than the federated performance, which is expected, where the separate performance seems to decrease with number of training images.

Table 10: Performance metrics of individual clients for VisDrone under different Dirichlet label skew settings. Separate mAP refers to local client performance on a subset of the training data, and federated mAP reflects performance after federated training.

Dirichlet α	Client index	Person samples	Vehicle samples	Separated mAP@.5	Federated mAP@.5
IID	Client 0	410	1208	0.53	0.56
	Client 1	410	1208	0.52	0.57
	Client 2	410	1208	0.53	0.56
	Client 3	410	1208	0.53	0.56
	Average			0.53 ± 0.01	0.56 ± 0.01
Dirichlet(0.5)	Client 0	361	193	0.51	0.55
	Client 1	16	2851	0.50	0.53
	Client 2	33	869	0.44	0.59
	Client 3	1231	869	0.55	0.57
	Average			0.50 ± 0.04	0.56 ± 0.02
Dirichlet(0.3)	Client 0	394	48	0.48	0.53
	Client 1	1214	4309	0.52	0.54
	Client 2	16	290	0.45	0.53
	Client 3	16	193	0.44	0.53
	Average			0.47 ± 0.03	0.53 ± 0.01
Dirichlet(0.1)	Client 0	16	4637	0.59	0.51
	Client 1	16	48	0.29	0.44
	Client 2	16	97	0.40	0.53
	Client 3	1592	48	0.49	0.56
	Average			0.44 ± 0.13	0.51 ± 0.04