Data Description and Exploration:

The dataset contains five different classes, as described in Table 1

Classes	Training Data Sample Points	Validation Data Sample Points	Testing Data Sample Points
Human	194	582	1940
Animal	194	582	
Truck	194	582	
Car	194	582	
Airplane	194	582	

Table 1. Statistics of the dataset

The dataset size is comparatively smaller compared to traditional datasets. The Train to Validation ratio is 1:3. Thus class imbalance problems can be excluded. However, the training size is significantly less for a model to extract a higher dimensional latent space.

The training dataset consists of classes rotated in a 3D space and is sheared to 2D; In contrast, the validation set is augmented with a permuted combination of the following transforms:

- Brightness
- Contrast
- Sharpness
- Rotation

However, to compensate for the 3D rotation space in a 2D dimensional space, rotation shearing was used. In terms of classes, classes "Truck" and "Car" resemble similar classes, which might confuse the model.

Model Exploration:

Considering the dataset's petite data points, a Siamese-Network approach was taken. A Conventional 10 Layers of CNN classifier was used to get a baseline of the data (performed with an accuracy of 28% [accuracy is measured on validation set]) with a 128-embedding dimension.

A Triplet Loss was used on the embeddings, while Cross-Entropy was used for the classification layer. Triplet Loss allows the model to compare two images and interpret whether they belong to the same class or not (i.e., data points from the same class are brought together, and two different classes are pushed further away in the latent space's distance and regularizing by using L2 norm).

This comparison of permutations of a batch of (same class, same class) and (same class, different class) allows the model to encounter more data as given in the dataset. This method allows the model to mitigate False Negatives and False Positives and at the same time enabling the model to learn the inherent features of each class from minimal data.

Adam Optimizer was used to solve the Gradient Descent with a CosineAnnealingLR Scheduler for the learning rates. Stochastic Weighted Averaging (SWA) was used on top of Adam to increase the efficiency of the learning and exploit the flatness of the training objectives while improving generalization.

Further, to scale up the dimensional space and to extract better features Residual Connection was adopted to transcend and preserve the features from a higher dimensional space to a lower-dimensional space. Residual Connection did aid the prediction accuracy to 31.1%.

Hence, a residual net backbone was selected for the Siamese network, with an embedding space of 64 dimensions and a block size [4, 4*4, 4*8]. This yielded 43.8% accuracy.

From the key takeaways from the data exploration, it is noticeable that the validation set is heavily augmented, and the variance causes the model to fail. Hence augmentations were adapted for training; this method did allow the model to generalize over the validation set and achieved 68% accuracy.

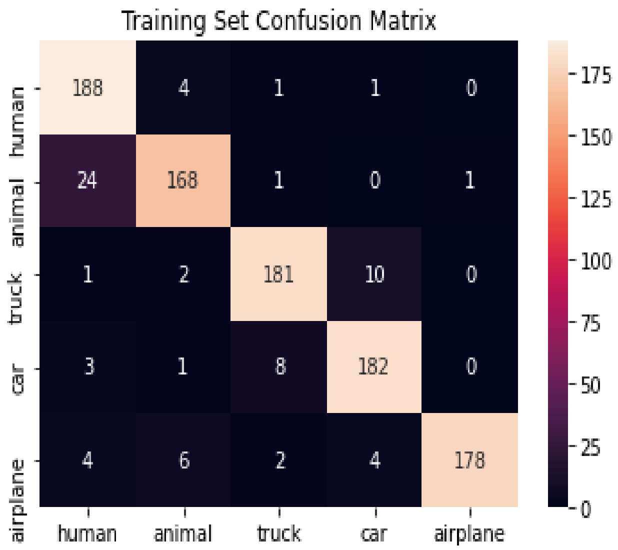
However, the brightness augmentor did brighten with contrast such that the image is not visible to the convolution network. To balance this a different contrast correcting augmentor was used in order to balance the ratio between the object and the background. Also the Cross-Entropy loss was replaced with Focal-Loss for the model to generalize the data and get a better hold of the underlying features. This allowed the model to understand the underlying variance and boosted the score to 58.4%.

But changing the augmentors value could only affect so much the model can comprehend. Hence a sweet spot for the augmentors was realized and but still, the model was not able to perform better than the previous setup. Therefore the layers were revamped to [4*2, 4*9, 4*12] with an embedding dimension of 64. This allowed the model to distill the underlying features with the help of the deeper layers of the network. Thus, boosting the accuracy of the network to 71%.

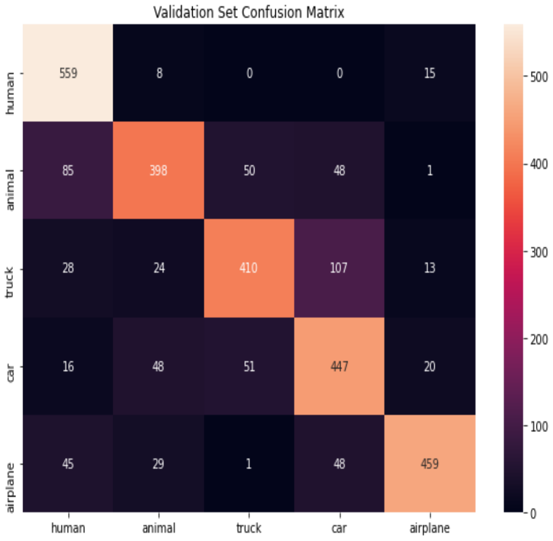
However, the model hit a roadblock beyond this point as increasing the model layers did not prove to improve the performance on the validation set, but the contrary made it worst. Since the other doors for improvement have been shut, this revealed that the embedding dimension of the network is not enough to compensate for the expansion of the network. Further increase in embedding space would saturate the model and would not affect the performance. Therefore, the embedding dimension was increased to 256 to accommodate the features to be represented in this latent space. This finally uplifted the final accuracy to **78.1%.** Given the size of the training dataset to be very small, the model has exhibited a very reasonable performance.

Results and Discussion:

Final validation accuracy of 78.1% was achieved over the validation dataset with a very small-scale dataset. In addition to this, the decision matrices were plotted for both Training and Validation set as plotted below in Fig 1,2.



1 Training Set Confusion Matrix



2 Validation Set Confusion Matrix

The accuracy for each class and for different sets are enlisted below:

Accuracy(%)/Classes	**Human**	**Animal**	**Truck**	**Car**	**Airplane**	**Overall**
Train	96.9	86.5	93.2	93.8	91.7	**92.42**
Validation	96.0	68.3	70.4	76.8	78.8	**78.1**

Due to time constraints, these explorations were refrained:

- Weighted Triplet Pair Combinations
  As predicted from Fig 1,2, it can be observed that the classes "Truck" and "Car" confuses the model and tried to classify each other as False Positives. While class "Humans" were getting classified as "Animals" the reason this occurs is since these two classes also share common features in them (ex. Horns of animal and sceptre of the human tribe). To eradicate this for the Triplet Function weighting was introduced to induce distinction between these two classes, which are being misclassified.

For example, while picking the pairs ([same-class, same-class], [same-class, different-class]) for the Training:

(i) Consider same-class to be "Car"; hence we perform a weighted random pick such that it picks the different-class "Truck" more often (increase the probability of picking the defined class)* and vice versa for the class "Truck"

(ii) Consider same-class to be "Human" then the model was focused on differentiating this with class "Animal" while the versa is not required as observed from the confusion matrix

**- 40% - For the class of interest (different-class)*

- The probability of picking same-class stays 0%

- The rest, 60%, was split into 20% equally for the rest of the 3 classes

- Contrastive Loss

  In replacement for the Triplet Loss, Contrastive Loss can be employed. This loss would not only solve for the distance but also intrinsically take the class into consideration. This might enable a better embedding learning for the Siamese Triplet Network compared to the Triplet Loss function.

- Ensemble of Models

  In addition to the above techniques, an ensemble of different backbones can be used in the model, and the best-performing model can be picked to crunch better accuracy.