Object Classification in CIFAR 10

Members:

Praneeth A S, Pratik Satapathy, Jash Dave, Jayant Golhar

CIFAR 10 Dataset

- Images database consisting of 10 different object classes
- Total 50000 images in the training set
- Total 10000 images in the test set

Train Dataset

Test Dataset

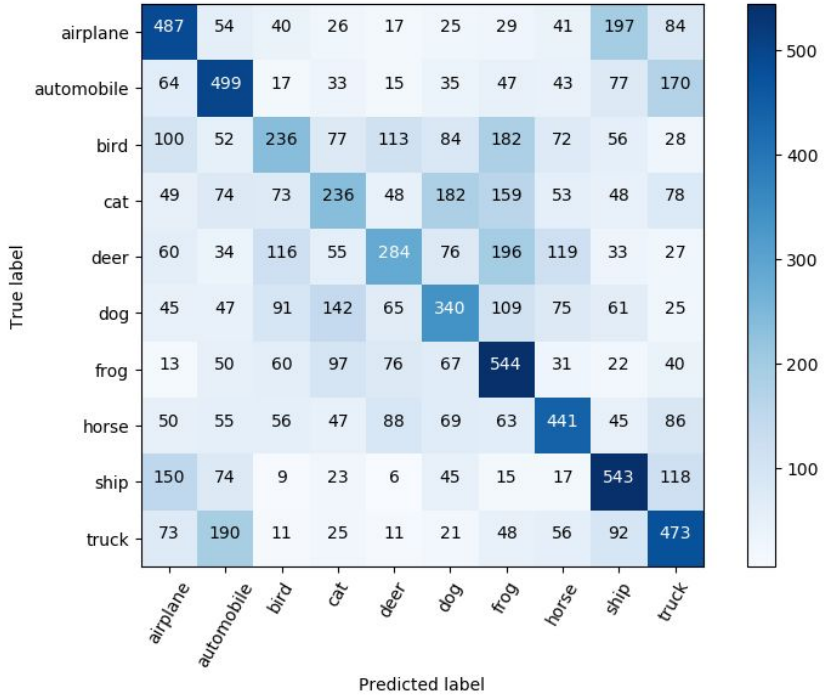
Linear classifier + Raw

Param and accuracy

Test accuracy : 0.4083

- Activation function : relu
- Gradient Descent Method : adam
- Learning rate type : constant
- Learning rate : 0.01
- Regularization constant : 0.01
- Error Tolerance : 1e-07
- Data Transformation method : StandardScaler
- Training epochs : 400

Confusion matrix



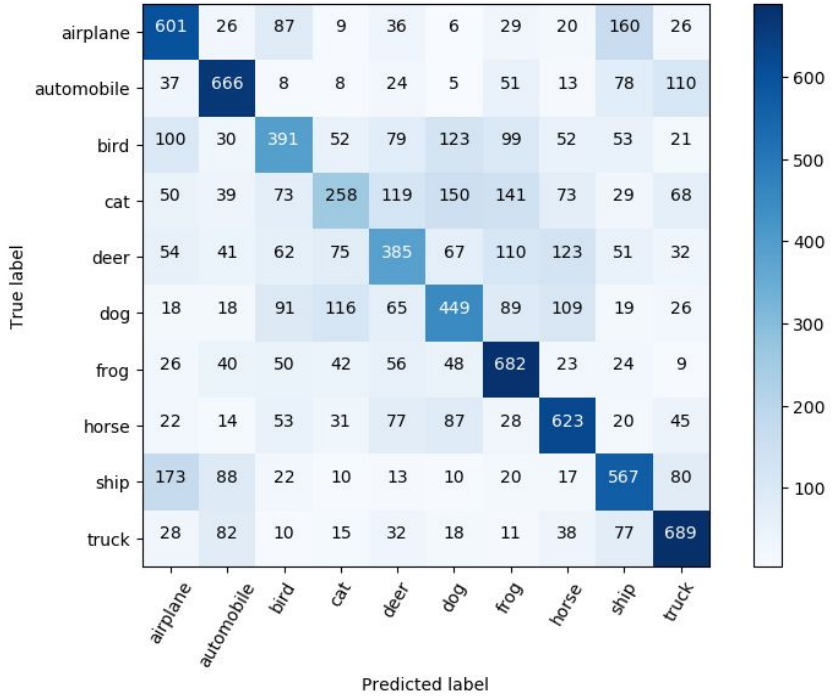
Linear classifier + HOG

Param and accuracy

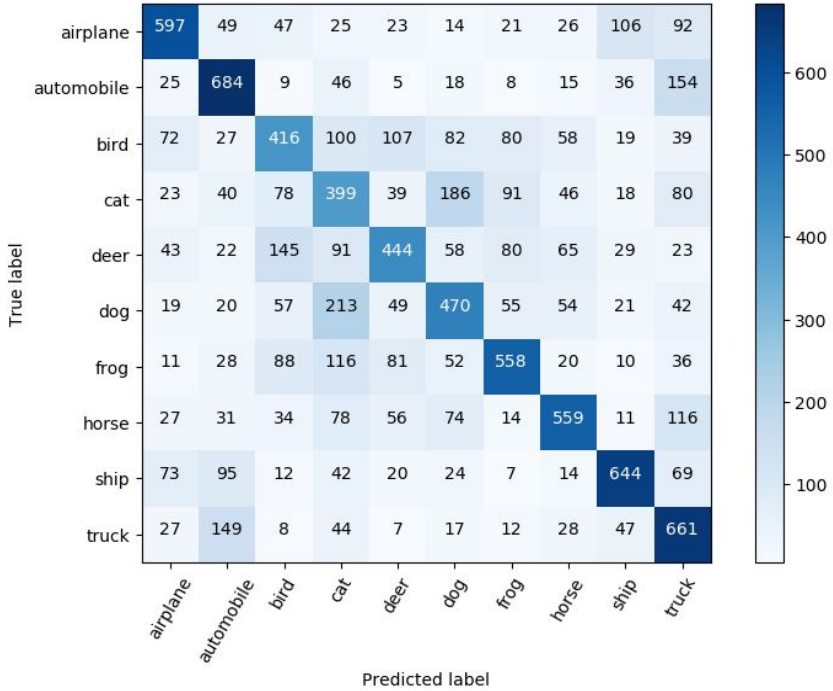
Test accuracy : 0.5311

- Activation function : relu
- Gradient Descent Method : gradient
- Learning rate type : constant
- Learning rate : 0.01
- Regularization constant : 0.01
- Error Tolerance : 1e-07
- Data Transformation method : StandardScaler
- Training epochs : 1000

Confusion matrix



SVM + Raw pixels

Param and accuracy	Confusion matrix
Test accuracy : 0.5432 ● Train validation split : 0.2 ● Best parameters ○ svm_kernel = 'rbf' ○ svm_gamma = 0.001 ○ svm_c = 100.0 ○ svm_max_iter = 10000 ○ svm_cs =1000} ● Accuracy score on test data = 0.5432 ● Trained using full train dataset ● Prediction done on 10000 images	

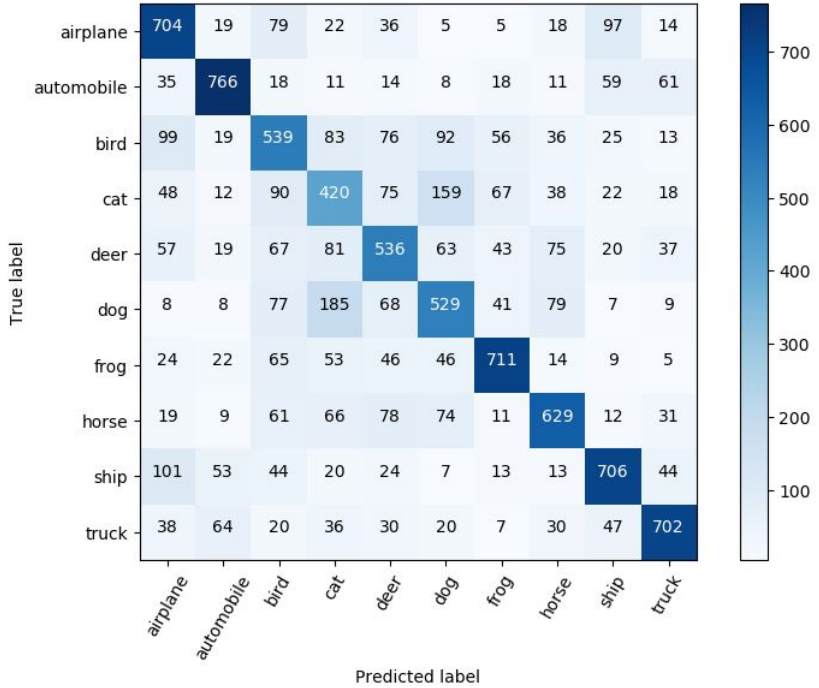
SVM + HOG

Param and accuracy

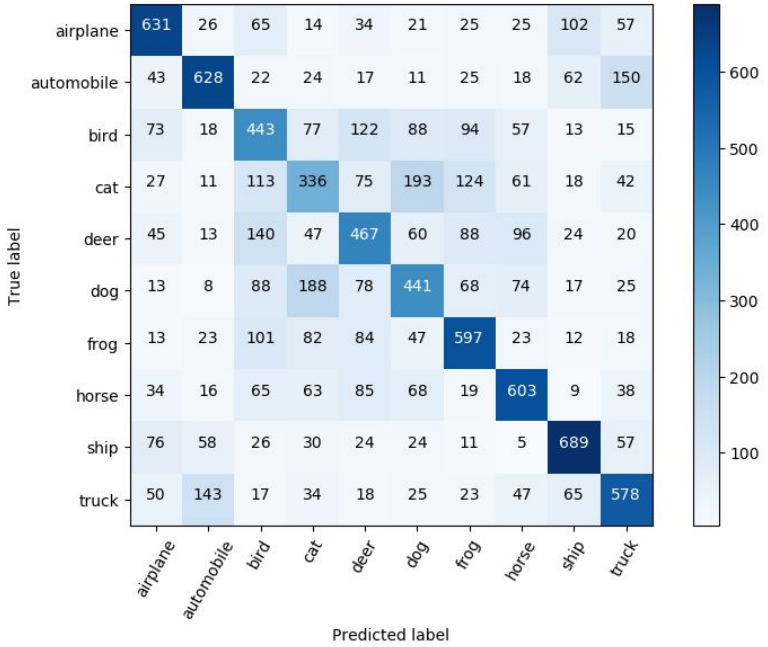
Test accuracy : 0.6095

- Train Validation split : 0.2
- Best parameters
  - svm_kernel = 'rbf'
  - svm_gamma = 0.001
  - svm_c = 1.0
  - svm_max_iter = 10000
  - svm_cs =1000
- Accuracy score on test data = 0.6095
- Trained using full train dataset
- Prediction done on 10000 images

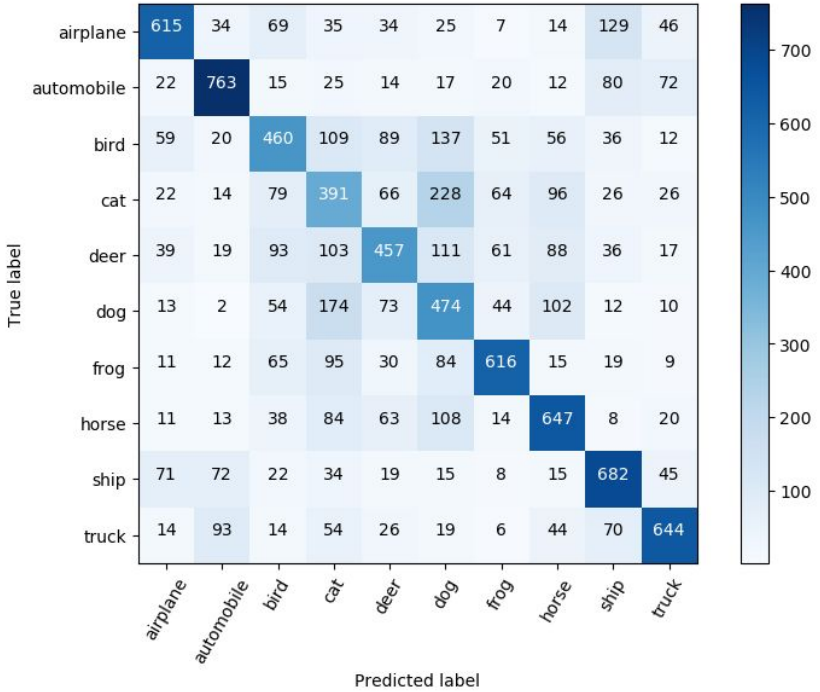
Confusion matrix



MLP + Raw pixels

Param and accuracy	Confusion matrix
Test accuracy : 0.5413 ● Train test split : 0.2 ○ Best parameters mlp_hidden_size = (3072,3072) ○ mlp_iterations = 200 ○ mlp_solver = 'adam' ○ Batch_size = 200 ● mlp_alpha = 1e-5 ● Test accuracy 0.5413 ● Trained using full train dataset ● Prediction done on 10000 images	

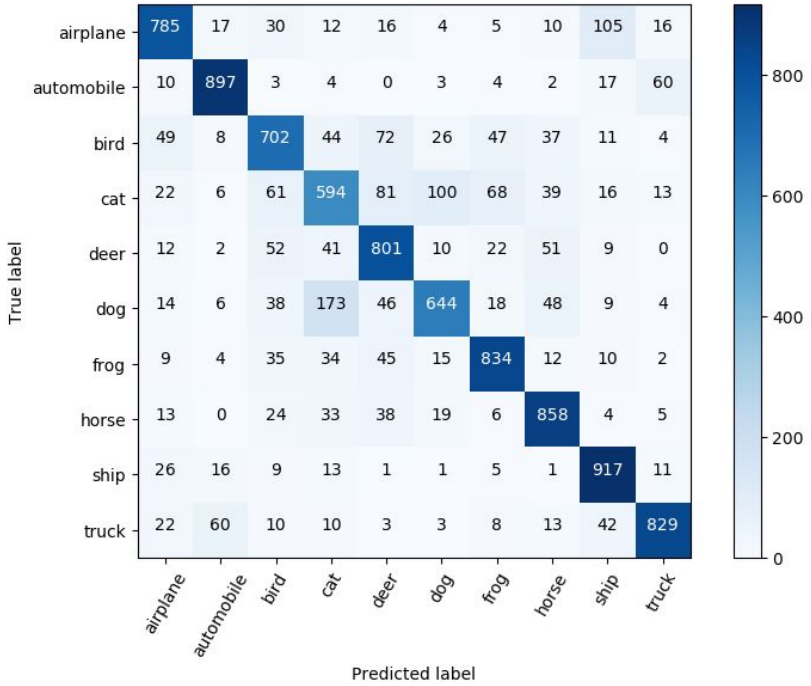
MLP + HOG features

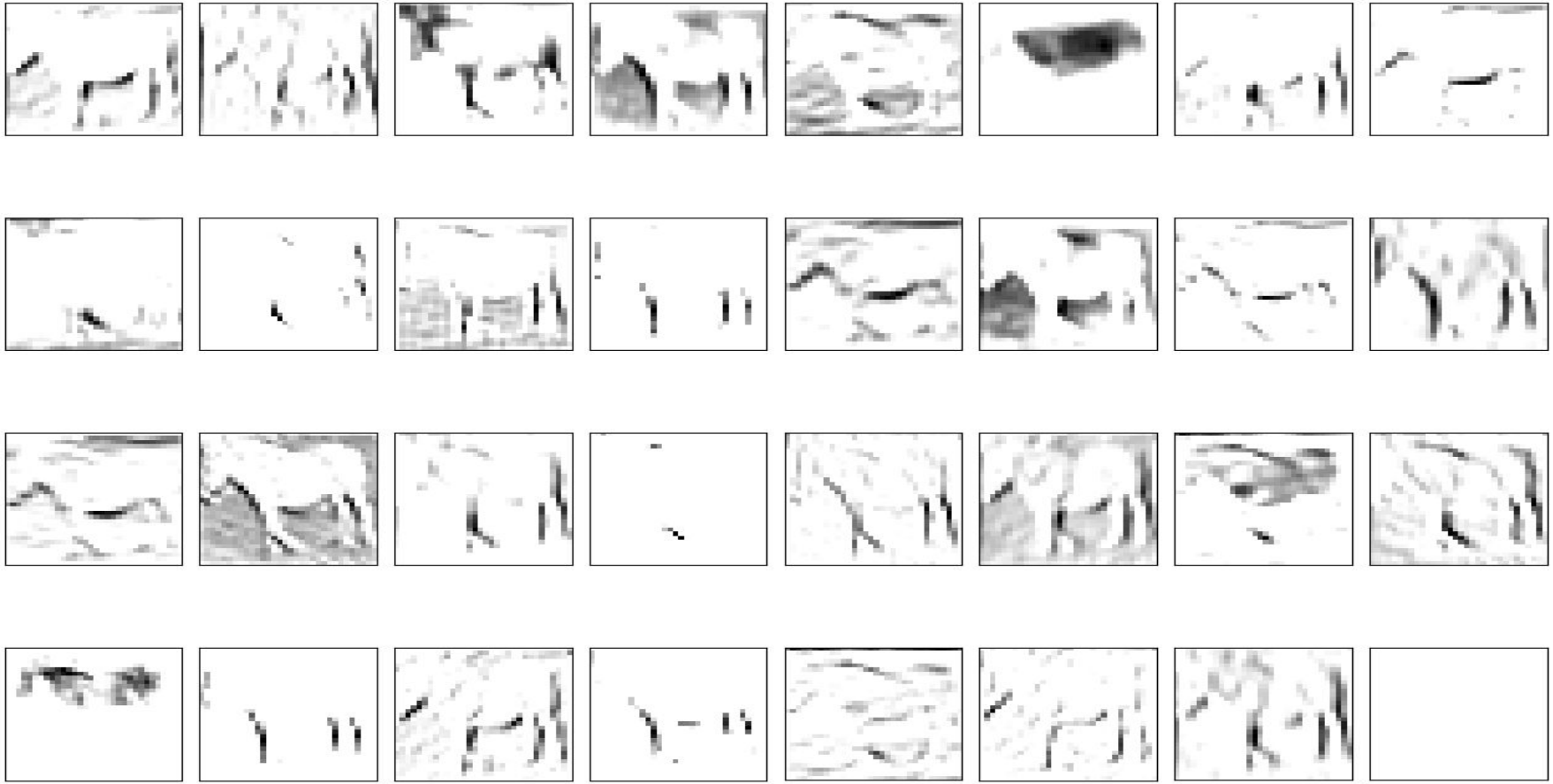
Param and accuracy	Confusion matrix
Test accuracy : 0.5621 ● Train test split : 0.2 ● Best parameters ○ mlp_hidden_size = (3072,3072) ○ mlp_iterations = 10000 ○ mlp_solver = 'adam' ○ mlp_alpha = 1e-5 ○ Batch_size = 200 ● Accuracy score on test data = 0.5621 ● Trained using full train dataset ● Prediction done on 10000 images	

Convolutional Neural Networks

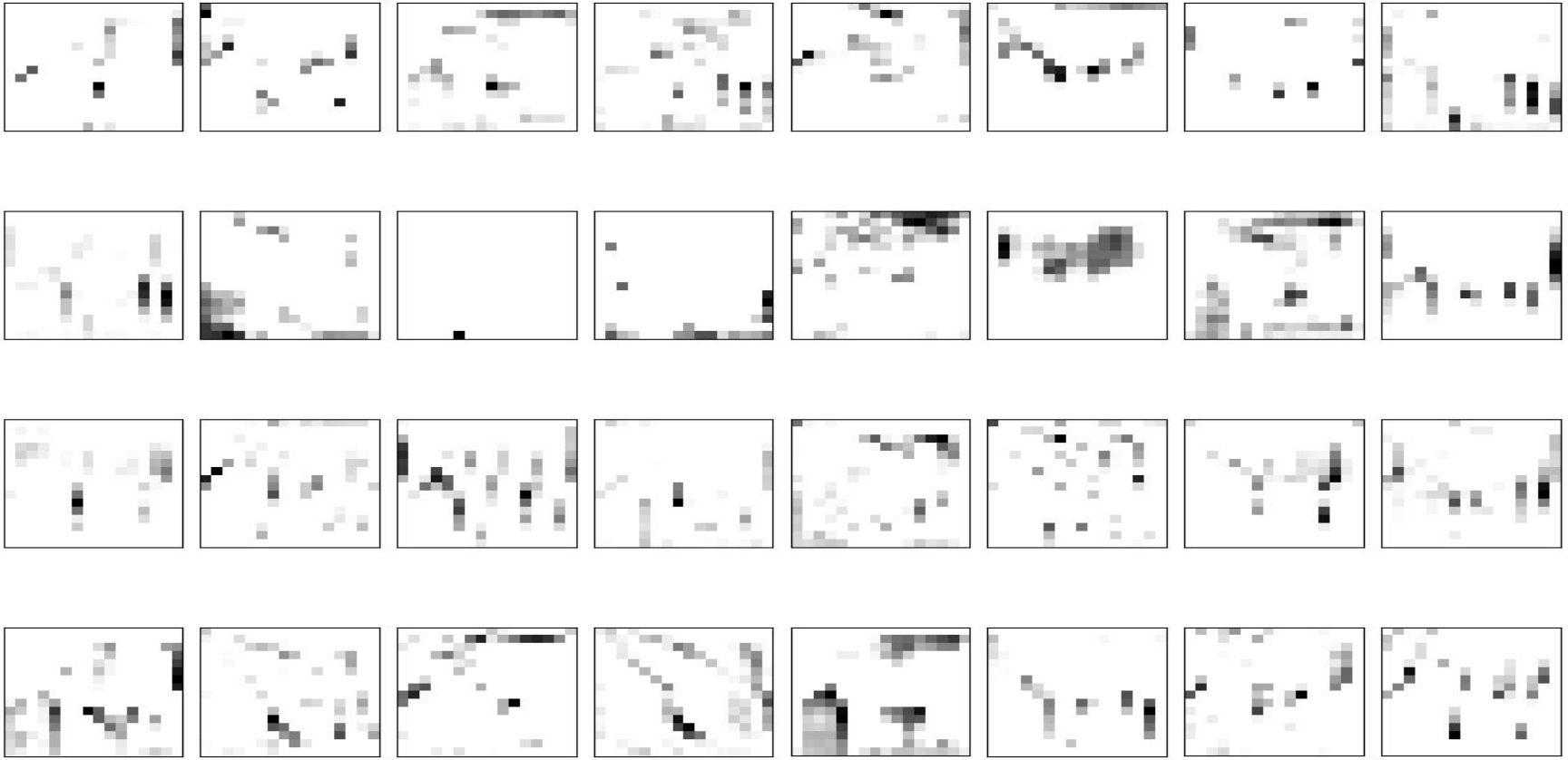
A CNN model with TFLearn

Param and accuracy	Architecture	Confusion matrix

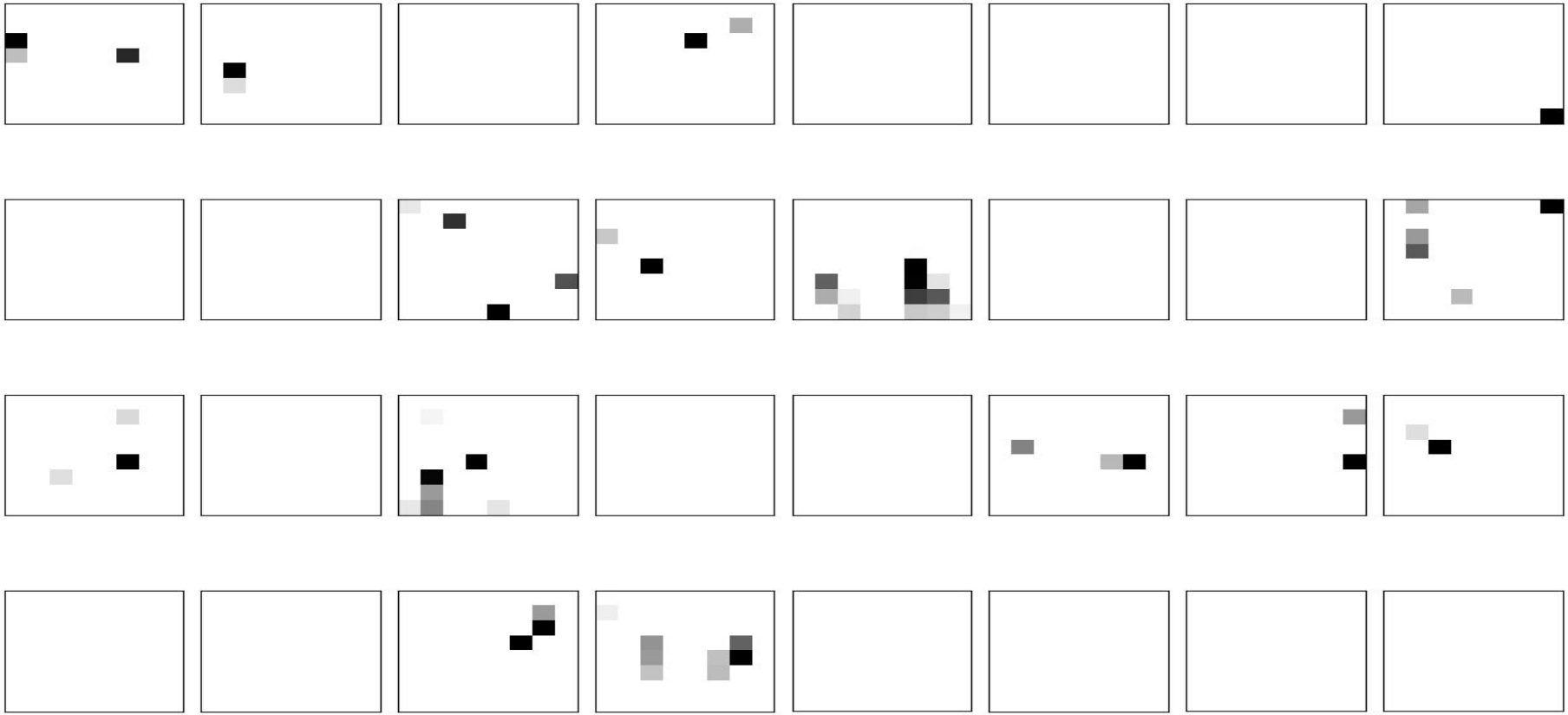
Test accuracy : 0.745
(without data augmentation)
 	
Test accuracy : 0.786
(with data augmentation)
 	
• Learning_rate=0.001
• Activations :
  ○ Relu at all layers
  ○ Softmax at o/p layer
• Optimizer='adam'
• Batch_size=96
• Epoch = 100
• Loss=categorical_crossentropy
• Feature engineering
  ○ Zero center
  ○ Standard norm
  ○ Rand flip left right
  ○ Rand rotation
	32 nos 3x3 conv layer
	Maxpooling layer 2x2
	64 nos 3x3 conv layer
	Maxpooling layer 2x2
	128 nos 3x3 conv layer
	Maxpooling layer 2x2
	256 nos 3x3 conv layer
	Maxpooling layer 2x2
	512 nos 3x3 conv layer
	Maxpooling layer 2x2
	
	Fully connected layer 1024
	Dropout 0.5
	Fully connected layer 2048
	Dropout 0.5
	Output layer 10
		



Convolution layer values for layer 1 for a horse image

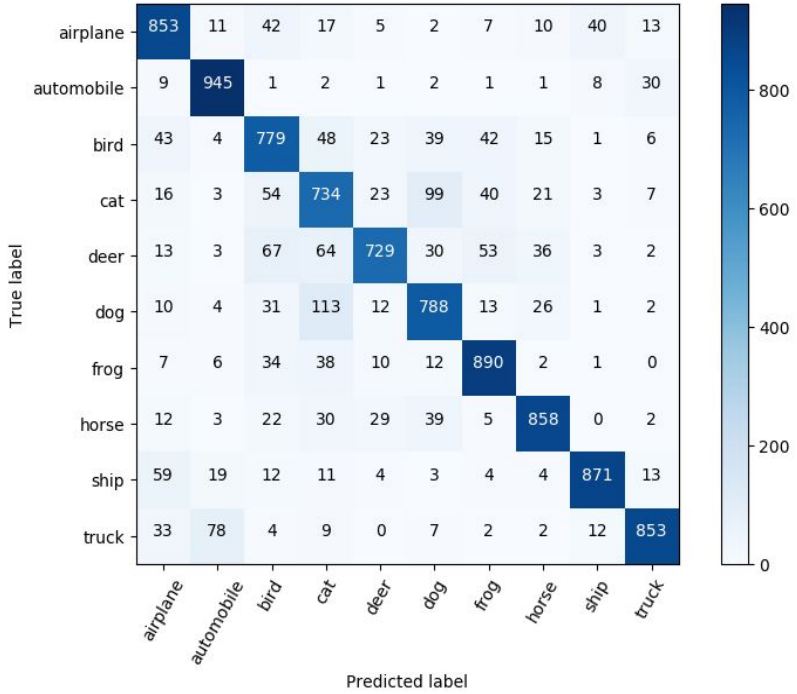


Convolution layer values for layer 2 for a horse image



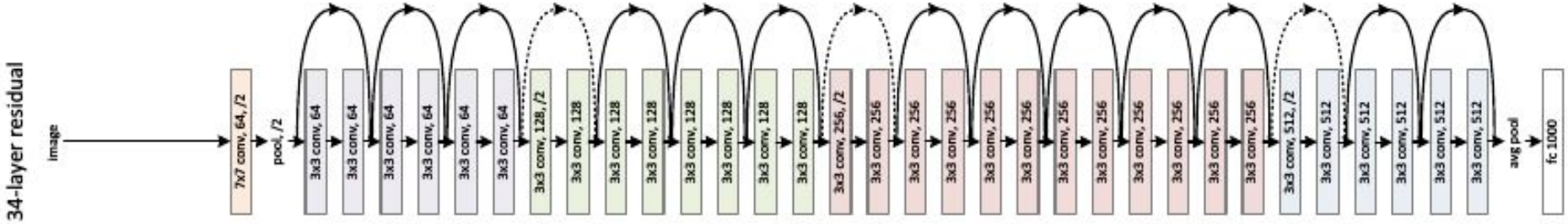
Convolution layer values for layer 3 for horse image

A CNN model (plain non residual design)

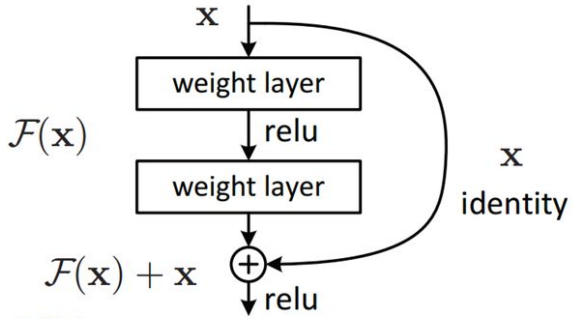
Param and accuracy	Architecture	Confusion matrix
Test accuracy : 0.83 ● Learning_rate=0.001 ● Activations : ○ Relu at all layers ○ Softmax at o/p layer ● Optimizer=adam ● Loss='categorical_crossentropy' ● Batch_size=96 ● Epoch = 100 ● Feature engineering ○ Zero center ○ Standard norm ○ Random flip left right	**32 nos 3x3 conv layer (4 layer)** **Max Pooling layer 2x2** **64 nos 3x3 conv layer (4 layer)** **Max Pooling layer 2x2** **Fully connected layer 512** **Dropout 0.5** **Output layer 10**	

Microsoft ResNet (winner of ILSVRC 2015)

Resnet architecture :

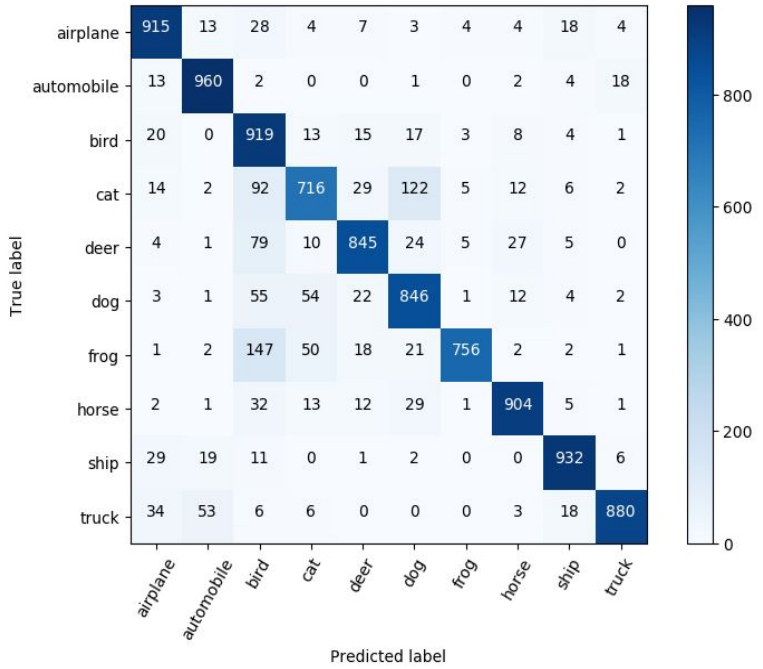


Residual block architecture :

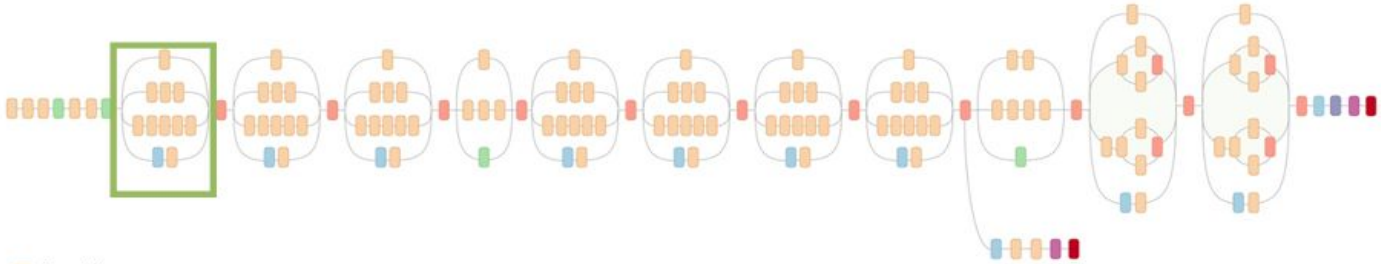


Ref: https://arxiv.org/pdf/1512.03385.pdf

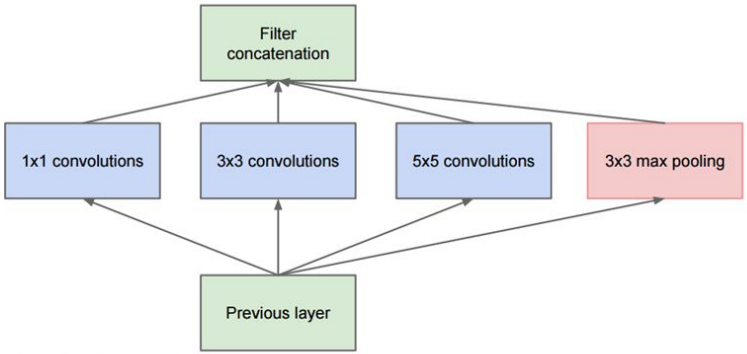
A smaller CNN inspired from Microsoft (ResNet)

Param and accuracy	Architecture	Confusion matrix
Test accuracy : 0.87 ● Learning_rate=0.001 ● Activations : ○ Relu at all layers ○ Softmax at o/p layer ● Optimizer=momentum ● Loss=categorical_crossentropy ● Batch_size=128 ● Epoch = 100 ● Feature engineering ○ Zero center ○ Standard norm ○ Ran flip left right ○ Rand rotation ○ Rand rotation/crop	**16 nos 3x3 conv layer** **5 residual block 16 o/p ch** **5 residual block 32 o/p ch** **5 residual block 64 o/p ch** **Relu activation** **Global avg pool** **Fully connected o/p layer 10**	

CNN transfer learning with Inception model

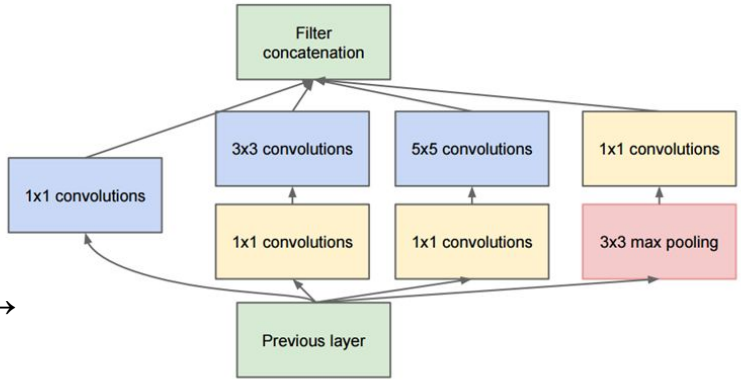


GoogLeNet Architecture



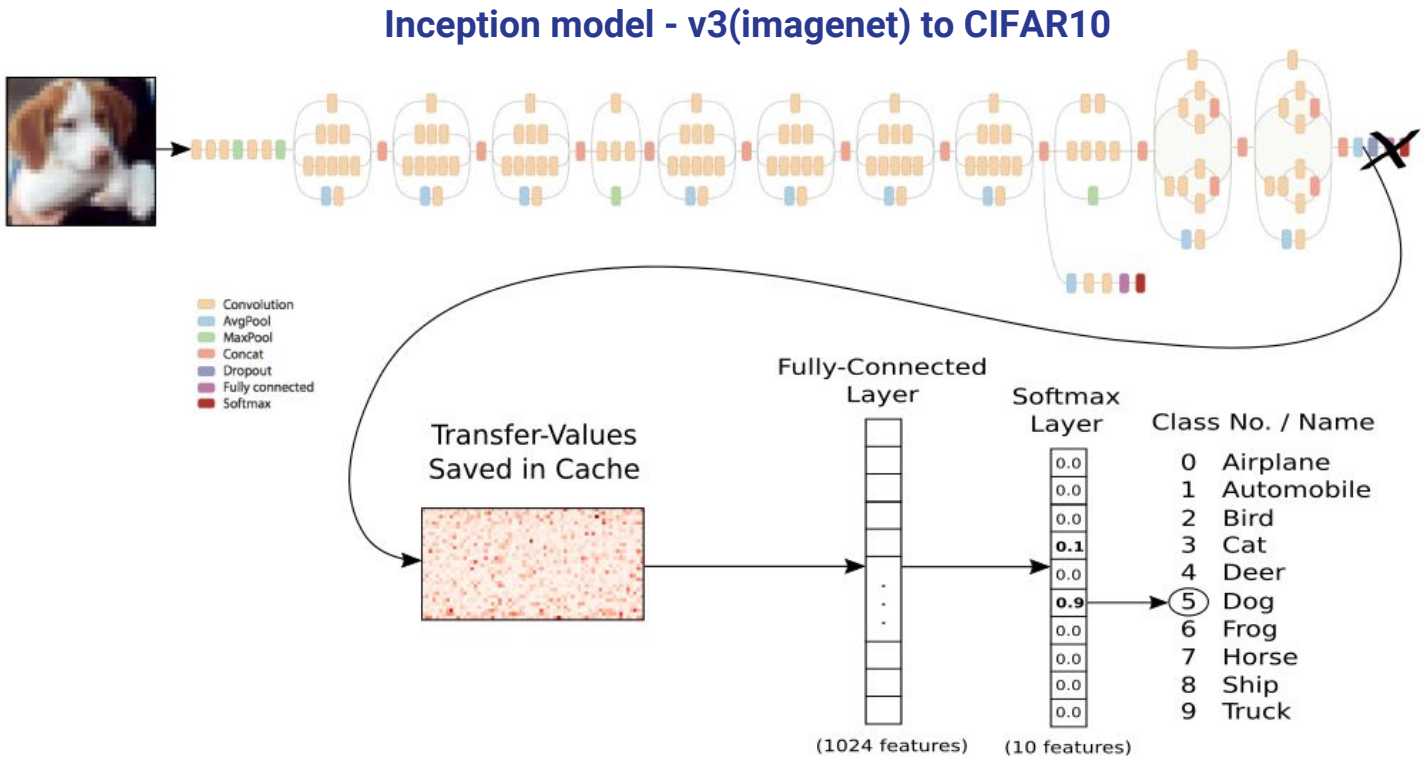
Naïve idea of an Inception module

← Building blocks →

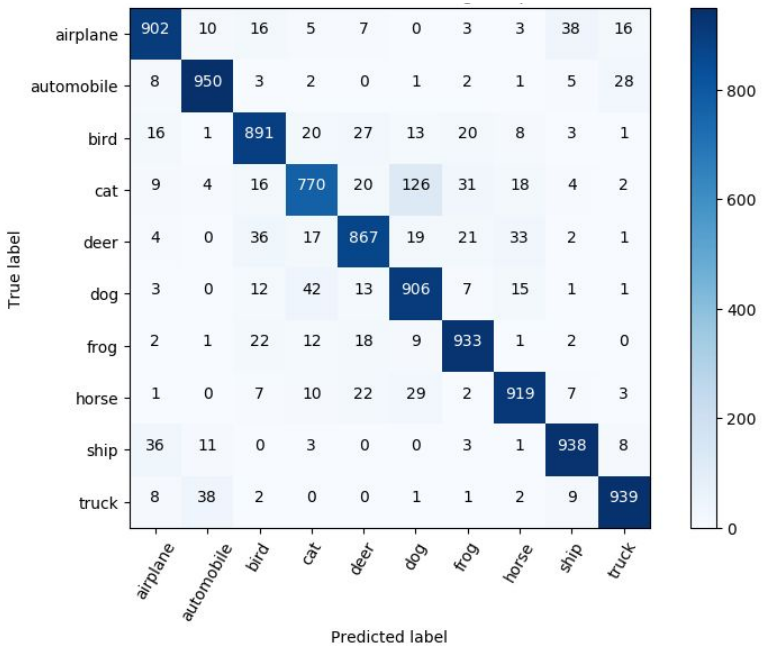


Full Inception module

Transfer Learning



Inception v3 transfer learning to CIFAR10

Param and accuracy	Architecture	Confusion matrix
Test accuracy : 0.9010 ● Learning_rate=0.0001 ● Activations : ○ Relu at all layers ○ Softmax at o/p layer ● Optimizer=adam ● Loss='categorical_crossentropy' ● Batch_size=256 ● Epoch = 10000	**Inception model** **(non trainable layers)** . . . **Fully connected layer 100** **(trainable layer)** **Fully connected layer 10** **(trainable layer)** **Softmax classifier**	

Summary of Experiments

Model#	Approach + Input data	Parameters (Activation function, training epochs, learning rate)	Training Accuracy	Validation Accuracy	Test Accuracy
1	Linear Classifier + Raw	Softmax, 400, 0.01	0.4243	0.4156	0.4083
2	Linear Classifier + HOG	Softmax, 1000, 0.01	0.5494	0.5324	0.5311
3	SVM + Raw	RBF, gamma=0.001, c=100, cs=1000	-	0.5353	0.5432
4	SVM + HOG	RBF, gamma=0.001, c=1, cs=1000	-	0.6127	0.6193
5	MLP + Raw	Sigmoid, 1000, 1.5, hidden: (3072, 3072)	1.0	0.5448	0.5413
6	MLP + HOG	alpha:1e-05, hidden: (3072, 3072), adam	-	0.5749	0.5621

Summary of Experiments (contd..)

Model#	Approach + Input data	Parameters (Activation function, training epochs, learning rate)	Training Accuracy	Validation Accuracy	Test Accuracy
7	CNN	Relu, 100, 0.001, optimizer: adam [model arch in slide : 12]	0.8926	0.7915	0.7861
8	CNN non residual	Relu, 100, 0.001, optimizer: adam [model arch in slide : 13]	0.9072	0.8362	0.8299
9	CNN Resnet inspired	Relu, 100, 0.001, optimizer: momentum [model arch in slide : 15]	0.9471	0.8684	0.8673
10	Transfer learning (Inception)	Relu, 10000, 0.0001, optimizer: adam [model arch in slide : 14]	0.9801	-	0.9015

References

1. Navneet Dalal, Bill Triggs - Histogram of Oriented Gradients - CVPR 2005
2. Corinna Cortes, Vladimir Vapnik - Support Vector Networks - Machine Learning Journal
3. CIFAR-10 - https://www.cs.toronto.edu/~kriz/cifar.html, last visited 27/04/2017
4. Tutorials for Inception model - https://github.com/Hvass-Labs/TensorFlow-Tutorials, last visited 27/04/2017
5. Inception paper : https://arxiv.org/pdf/1409.4842v1.pdf
6. http://tflearn.org/
7. https://www.tensorflow.org/
8. Resnet https://arxiv.org/pdf/1512.03385.pdf
9. https://github.com/tflearn/tflearn repository
10. Alexnet https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf