Appendix A:

Table A.1: Test accuracy of machine learning classifiers

Machine Learning Classifier	Parameters	AstroSpectra-MNIST-v1	AstroSpectra-MNIST-v2	Fashion-MNIST
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 50, "splitter": "random"	0.778	0.724	0.794
DecisionTreeClassifier	"criterion": "gini", "max_depth": 10, "splitter": "random"	0.774	0.745	0.795
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 10, "splitter": "random"	0.784	0.734	0.792
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 100 ,"splitter": "random"	0.780	0.728	0.794
DecisionTreeClassifier	"criterion": "gini", "max_depth": 50 ,"splitter": "random"	0.768	0.711	0.784
DecisionTreeClassifier	"criterion": "gini" ,"max_depth": 100, "splitter": "random"	0.764	0.713	0.784
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 10, "splitter": "best"	0.782	0.742	0.801
DecisionTreeClassifier	"criterion": "gini", "max_depth": 10 ,"splitter": "best"	0.795	0.742	0.799
DecisionTreeClassifier	"criterion": "gini", "max_depth": 100, "splitter": "best"	0.771	0.715	0.789
DecisionTreeClassifier	"criterion": "gini", "max_depth": 50, "splitter": "best"	0.768	0.717	0.789
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 100 ,"splitter": "best"	0.758	0.727	0.797
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 50, "splitter": "best"	0.750	0.729	0.797
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 10, "splitter": "random"	0.755	0.699	0.734
ExtraTreeClassifier	"criterion": "gini", "max_depth": 10 , "splitter": "random"	0.787	0.680	0.729
ExtraTreeClassifier	"criterion": "gini", "max_depth": 10, "splitter": "best"	0.763	0.715	0.776
ExtraTreeClassifier	"criterion": "gini", "max_depth": 50, "splitter": "random"	0.755	0.693	0.752
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 100, "splitter": "random"	0.754	0.694	0.751
ExtraTreeClassifier	"criterion": "gini", "max_depth": 100, "splitter": "random"	0.756	0.687	0.753
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 50, "splitter": "random"	0.772	0.697	0.756
ExtraTreeClassifier	"criterion": "gini", "max_depth": 100, "splitter": "best"	0.750	0.695	0.771
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 10, "splitter": "best"	0.782	0.704	0.780
ExtraTreeClassifier	"criterion": "gini", "max_depth": 50, "splitter": "best"	0.753	0.694	0.771
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 50, "splitter": "best"	0.776	0.695	0.778
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 100, "splitter": "best"	0.731	0.701	0.779
GaussianNB	"priors": [1/classnum, 1/classnum, 1/classnum]	0.642	0.597	0.564
GradientBoostingClassifier	loss="log_loss", "max_depth"=10, "n_estimators"=100	0.877	0.806	0.888
GradientBoostingClassifier	loss="log_loss", "max_depth"=50, "n_estimators"=10	0.780	0.742	0.804
GradientBoostingClassifier	loss="log_loss", "max_depth"=50, "n_estimators"=50	0.786	0.740	0.836
GradientBoostingClassifier	loss="log_loss", "max_depth"=50, "n_estimators"=100	0.779	0.740	0.838
KNeighborsClassifier	"n_neighbors": 1, "p": 2, "weights": "uniform"	0.812	0.674	0.847
KNeighborsClassifier	"n_neighbors": 9, "p": 1, "weights": "uniform"	0.853	0.730	0.856
KNeighborsClassifier	"n_neighbors": 5, "p": 1, "weights": "distance"	0.852	0.723	0.860
LinearSVC	"C": 1, "loss": "squared_hinge", "multi_class": "ovr" "penalty": "l2"	0.806	0.733	0.827
LinearSVC	"C": 1, "loss": "hinge", "multi_class": "ovr", "penalty": "l2"	0.802	0.739	0.837
LinearSVC	"C": 10, "loss": "hinge", "multi_class": "ovr", "penalty": "l2"	0.772	0.696	0.783
LinearSVC	"C": 100, "loss": "squared_hinge", "multi_class": "ovr", "penalty": "l2"	0.805	0.709	0.788
LinearSVC	"C": 10, "loss": "squared_hinge", "multi_class": "ovr", "penalty": "l2"	0.806	0.689	0.793
LinearSVC	"C"=100,"loss"="hinge","multi_class"="crammer_singer","penalty"="l2"	0.712	0.718	0.466
LinearSVC	"C"=100,"loss"="squared_hinge","multi_class"="crammer_singer","penalty"="l2"	0.713	0.718	0.458
LinearSVC	"C"=100,"loss"="hinge","multi_class"="crammer_singer", "penalty"="l1"	0.695	0.703	0.460
LinearSVC	"C"=100,"loss"="squared_hinge","multi_class"="crammer_singer","penalty"="l1"	0.704	0.708	0.444
LogisticRegression	"C":100,"multi_class":"ovr","penalty":"l2"	0.808	0.741	0.838
MLPClassifier	"activation": "relu", "hidden_layer_sizes": [10]	0.840	0.765	0.850
MLPClassifier	"activation": "relu", "hidden_layer_sizes": [10 10]	0.829	0.773	0.851
MLPClassifier	"activation": "tanh", "hidden_layer_sizes": [10]	0.826	0.756	0.841
MLPClassifier	"activation": "tanh", "hidden_layer_sizes": [10 10]	0.825	0.768	0.840
MLPClassifier	"activation": "relu", "hidden_layer_sizes": [100 10]	0.874	0.791	0.874
MLPClassifier	"activation": "relu", "hidden_layer_sizes": [100]	0.872	0.782	0.877
MLPClassifier	"activation": "tanh", "hidden_layer_sizes": [100 10]	0.864	0.775	0.865
MLPClassifier	"activation": "tanh", "hidden_layer_sizes": [100]	0.867	0.785	0.870
PassiveAggressiveClassifier	"C": 1	0.776	0.713	0.791
PassiveAggressiveClassifier	"C": 10	0.767	0.712	0.794
PassiveAggressiveClassifier	"C": 100	0.794	0.708	0.782

Table A.1: Test accuracy of machine learning classifiers

Machine Learning Classifier	**Parameters**	**AstroSpectra-MNIST-v1**	**AstroSpectra-MNIST-v2**	**Fashion-MNIST**
Perceptron	"penalty": "l2"	0.762	0.706	0.753
Perceptron	"penalty": "l1"	0.792	0.720	0.771
Perceptron	"penalty": "elasticnet"	0.751	0.692	0.759
RandomForestClassifier	"criterion": "gini", "max_depth": 10, "n_estimators": 10	0.836	0.751	0.834
RandomForestClassifier	"criterion": "gini", "max_depth": 100, "n_estimators": 10	0.847	0.752	0.851
RandomForestClassifier	"criterion": "entropy", "max_depth": 10, "n_estimators": 10	0.836	0.748	0.837
RandomForestClassifier	"criterion": "gini", "max_depth": 50, "n_estimators": 10	0.843	0.757	0.850
RandomForestClassifier	"criterion": "entropy", "max_depth": 50, "n_estimators": 10	0.834	0.755	0.853
RandomForestClassifier	"criterion": "entropy", "max_depth": 100, "n_estimators": 10	0.862	0.763	0.856
RandomForestClassifier	"criterion": "gini", "max_depth": 10, "n_estimators": 50	0.835	0.760	0.843
RandomForestClassifier	"criterion": "entropy", "max_depth": 10, "n_estimators": 50	0.836	0.757	0.845
RandomForestClassifier	"criterion": "gini", "max_depth": 100, "n_estimators": 50	0.872	0.780	0.873
RandomForestClassifier	"criterion": "gini", "max_depth": 50, "n_estimators": 50	0.871	0.781	0.873
RandomForestClassifier	"criterion": "gini", "max_depth": 10, "n_estimators": 100	0.854	0.758	0.844
RandomForestClassifier	"criterion": "gini", "max_depth": 100, "n_estimators": 100	0.870	0.784	0.876
RandomForestClassifier	"criterion": "entropy", "max_depth": 100, "n_estimators": 50	0.873	0.780	0.875
RandomForestClassifier	"criterion": "entropy", "max_depth": 50, "n_estimators": 50	0.867	0.785	0.874
RandomForestClassifier	"criterion": "entropy", "max_depth": 10, "n_estimators": 100	0.846	0.759	0.846
RandomForestClassifier	"criterion": "entropy", "max_depth": 100, "n_estimators": 100	0.878	0.785	0.877
RandomForestClassifier	"criterion": "entropy", "max_depth": 50, "n_estimators": 100	0.869	0.785	0.879
RandomForestClassifier	"criterion": "gini", "max_depth": 50, "n_estimators": 100	0.873	0.782	0.876
SGDClassifier	"loss": "squared_hinge", "penalty": "l2"	0.793	0.720	0.826
SGDClassifier	"loss": "hinge", "penalty": "l2"	0.800	0.728	0.826
SGDClassifier	"loss": "modified_huber", "penalty": "l2"	0.802	0.719	0.827
SGDClassifier	"loss": "perceptron", "penalty": "l2"	0.780	0.711	0.826
SGDClassifier	"loss": "hinge", "penalty": "elasticnet"	0.791	0.727	0.829
SGDClassifier	"loss": "log_loss", "penalty": "l2"	0.777	0.735	0.827
SGDClassifier	"loss": "perceptron", "penalty": "l1"	0.772	0.732	0.819
SGDClassifier	"loss": "squared_hinge", "penalty": "l1"	0.771	0.694	0.819
SGDClassifier	"loss": "modified_huber", "penalty": "elasticnet"	0.792	0.722	0.831
SGDClassifier	"loss": "squared_hinge", "penalty": "elasticnet"	0.783	0.715	0.828
SGDClassifier	"loss": "log_loss", "penalty": "l1"	0.775	0.737	0.821
SGDClassifier	"loss": "hinge", "penalty": "l1"	0.775	0.735	0.820
SGDClassifier	"loss": "modified_huber", "penalty": "l1"	0.767	0.736	0.821
SGDClassifier	"loss": "perceptron", "penalty": "elasticnet"	0.769	0.724	0.828
SGDClassifier	"loss": "log_loss", "penalty": "elasticnet"	0.782	0.741	0.826

Appendix B:

Table B.1: Machine learning test results and comparison for LineAstroSpectra

Machine Learning Classifier	**Parameters**	**LineAstro Spectra-v1**	**AstroSpectra-MNIST-v1**	**LineAstro Spectra-v2**	**AstroSpectra-MNIST-v2**
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 50, "splitter": "random"	0.795	0.778	0.649	0.724
DecisionTreeClassifier	"criterion": "gini", "max_depth": 10, "splitter": "random"	0.779	0.774	0.557	0.745
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 10, "splitter": "random"	0.793	0.784	0.591	0.734
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 100, "splitter": "random"	0.803	0.780	0.646	0.728
DecisionTreeClassifier	"criterion": "gini", "max_depth": 50, "splitter": "random"	0.807	0.768	0.630	0.711
DecisionTreeClassifier	"criterion": "gini", "max_depth": 100, "splitter": "random"	0.814	0.764	0.658	0.713
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 10, "splitter": "best"	0.807	0.782	0.643	0.742
DecisionTreeClassifier	"criterion": "gini", "max_depth": 10, "splitter": "best"	0.795	0.795	0.657	0.742
DecisionTreeClassifier	"criterion": "gini", "max_depth": 100, "splitter": "best"	0.789	0.771	0.656	0.715
DecisionTreeClassifier	"criterion": "gini", "max_depth": 50, "splitter": "best"	0.803	0.768	0.651	0.717
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 100, "splitter": "best"	0.810	0.758	0.671	0.727
DecisionTreeClassifier	"criterion": "entropy", "max_depth": 50, "splitter": "best"	0.809	0.750	0.666	0.729
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 10, "splitter": "random"	0.664	0.755	0.512	0.699
ExtraTreeClassifier	"criterion": "gini", "max_depth": 10, "splitter": "random"	0.661	0.787	0.507	0.680
ExtraTreeClassifier	"criterion": "gini", "max_depth": 10, "splitter": "best"	0.769	0.763	0.603	0.715
ExtraTreeClassifier	"criterion": "gini", "max_depth": 50, "splitter": "random"	0.762	0.755	0.606	0.693
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 100, "splitter": "random"	0.789	0.754	0.616	0.694
ExtraTreeClassifier	"criterion": "gini", "max_depth": 100, "splitter": "random"	0.778	0.756	0.604	0.687
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 50, "splitter": "random"	0.769	0.772	0.595	0.697
ExtraTreeClassifier	"criterion": "gini", "max_depth": 100, "splitter": "best"	0.795	0.750	0.619	0.695
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 10, "splitter": "best"	0.783	0.782	0.626	0.704
ExtraTreeClassifier	"criterion": "gini", "max_depth": 50, "splitter": "best"	0.794	0.753	0.628	0.694
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 50, "splitter": "best"	0.786	0.776	0.640	0.695
ExtraTreeClassifier	"criterion": "entropy", "max_depth": 100, "splitter": "best"	0.792	0.731	0.626	0.701
GaussianNB	"priors": [1/classnum, 1/classnum, 1/classnum]	0.359	0.642	0.303	0.597
GradientBoostingClassifier	"loss"="log_loss", "max_depth"=10, "n_estimators"=100	0.899	0.877	0.735	0.806
GradientBoostingClassifier	"loss"="log_loss", "max_depth"=50, "n_estimators"=10	0.829	0.780	0.678	0.742
GradientBoostingClassifier	"loss"="log_loss", "max_depth"=50, "n_estimators"=50	0.826	0.786	0.674	0.740
GradientBoostingClassifier	"loss"="log_loss", "max_depth"=50, "n_estimators"=100	0.828	0.779	0.674	0.740
KNeighborsClassifier	"n_neighbors": 1, "p": 2, "weights": "uniform"	0.760	0.812	0.599	0.674
KNeighborsClassifier	"n_neighbors": 9, "p": 1, "weights": "uniform"	0.737	0.853	0.608	0.730
KNeighborsClassifier	"n_neighbors": 5, "p": 1, "weights": "distance"	0.741	0.852	0.613	0.723
LinearSVC	"C": 1, "loss": "squared_hinge", "multi_class": "ovr", "penalty": "l2"	0.457	0.806	0.640	0.733
LinearSVC	"C": 1,"loss":"hinge","multi_class": "ovr","penalty":"l2"	0.535	0.802	0.638	0.739
LinearSVC	"C": 10,"loss":"hinge","multi_class": "ovr","penalty":"l2"	0.522	0.772	0.694	0.696
LinearSVC	"C": 100,"loss":"squared_hinge","multi_class":"ovr","penalty": "l2"	0.511	0.805	0.728	0.709
LinearSVC	"C": 10,"loss":"squared_hinge","multi_class":"ovr","penalty": "l2"	0.525	0.806	0.698	0.689
LinearSVC	"C"=100,"loss"="hinge","multi_class"="crammer_singer","penalty"="l2"	0.664	0.712	0.713	0.718
LinearSVC	"C":100,"loss":"squared_hinge","multi_class":"crammer_singer", "penalty":"l2"	0.665	0.713	0.668	0.718
LinearSVC	"C"=100,"loss"="hinge","multi_class"="crammer_singer","penalty"="l1"	0.665	0.695	0.666	0.703
LinearSVC	"C":100,"loss":"squared_hinge","multi_class":"crammer_singer", "penalty":"l1"	0.665	0.704	0.732	0.708
LogisticRegression	"C":100,"multi_class":"ovr","penalty":"l2"	0.524	0.808	0.657	0.741
MLPClassifier	"activation": "relu", "hidden_layer_sizes": [10]	0.572	0.840	0.721	0.765
MLPClassifier	"activation": "relu", "hidden_layer_sizes": [10 10]	0.618	0.829	0.720	0.773
MLPClassifier	"activation": "tanh", "hidden_layer_sizes": [10]	0.554	0.826	0.698	0.756
MLPClassifier	"activation": "tanh", "hidden_layer_sizes": [10 10]	0.554	0.825	0.718	0.768
MLPClassifier	"activation": "relu", "hidden_layer_sizes": [100 10]	0.590	0.874	0.777	0.791
MLPClassifier	"activation": "relu", "hidden_layer_sizes": [100]	0.553	0.872	0.698	0.782
MLPClassifier	"activation": "tanh", "hidden_layer_sizes": [100 10]	0.560	0.864	0.787	0.775
MLPClassifier	"activation": "tanh", "hidden_layer_sizes": [100]	0.530	0.867	0.732	0.785
PassiveAggressiveClassifier	"C": 1	0.515	0.776	0.692	0.713

Table B.1: Machine learning test results and comparison for LineAstroSpectra

Machine Learning Classifier	**Parameters**	**LineAstro Spectra-v1**	**AstroSpectra-MNIST-v1**	**LineAstro Spectra-v2**	**AstroSpectra-MNIST-v2**
PassiveAggressiveClassifier	"C": 10	0.343	0.767	0.627	0.712
PassiveAggressiveClassifier	"C": 100	0.431	0.794	0.646	0.708
Perceptron	"penalty": "l2"	0.512	0.762	0.575	0.706
Perceptron	"penalty": "l1"	0.506	0.792	0.579	0.720
Perceptron	"penalty": "elasticnet"	0.437	0.751	0.499	0.692
RandomForestClassifier	"criterion": "gini", "max_depth": 10, "n_estimators": 10	0.834	0.836	0.629	0.751
RandomForestClassifier	"criterion": "gini", "max_depth": 100, "n_estimators": 10	0.862	0.847	0.676	0.752
RandomForestClassifier	"criterion": "entropy", "max_depth": 10, "n_estimators": 10	0.845	0.836	0.642	0.748
RandomForestClassifier	"criterion": "gini", "max_depth": 50, "n_estimators": 10	0.862	0.843	0.674	0.757
RandomForestClassifier	"criterion": "entropy", "max_depth": 50, "n_estimators": 10	0.864	0.834	0.681	0.755
RandomForestClassifier	"criterion": "entropy", "max_depth": 100, "n_estimators": 10	0.833	0.862	0.684	0.763
RandomForestClassifier	"criterion": "gini", "max_depth": 10, "n_estimators": 50	0.847	0.835	0.639	0.760
RandomForestClassifier	"criterion": "entropy", "max_depth": 10, "n_estimators": 50	0.848	0.836	0.648	0.757
RandomForestClassifier	"criterion": "gini", "max_depth": 100, "n_estimators": 50	0.875	0.872	0.711	0.780
RandomForestClassifier	"criterion": "gini", "max_depth": 50, "n_estimators": 50	0.877	0.871	0.708	0.781
RandomForestClassifier	"criterion": "gini", "max_depth": 10, "n_estimators": 100	0.846	0.854	0.641	0.758
RandomForestClassifier	"criterion": "gini", "max_depth": 100, "n_estimators": 100	0.882	0.870	0.715	0.784
RandomForestClassifier	"criterion": "entropy", "max_depth": 100, "n_estimators": 50	0.881	0.873	0.704	0.780
RandomForestClassifier	"criterion": "entropy", "max_depth": 50, "n_estimators": 50	0.878	0.867	0.706	0.785
RandomForestClassifier	"criterion": "entropy", "max_depth": 10, "n_estimators": 100	0.861	0.846	0.646	0.759
RandomForestClassifier	"criterion": "entropy", "max_depth": 100, "n_estimators": 100	0.877	0.878	0.712	0.785
RandomForestClassifier	"criterion": "entropy", "max_depth": 50, "n_estimators": 100	0.883	0.869	0.715	0.785
RandomForestClassifier	"criterion": "gini", "max_depth": 50, "n_estimators": 100	0.881	0.873	0.719	0.782
SGDClassifier	"loss": "squared_hinge", "penalty": "l2"	0.522	0.793	0.679	0.720
SGDClassifier	"loss": "hinge", "penalty": "l2"	0.523	0.800	0.587	0.728
SGDClassifier	"loss": "modified_huber", "penalty": "l2"	0.499	0.802	0.534	0.719
SGDClassifier	"loss": "perceptron", "penalty": "l2"	0.540	0.780	0.552	0.711
SGDClassifier	"loss": "hinge", "penalty": "elasticnet"	0.480	0.791	0.575	0.727
SGDClassifier	"loss": "log_loss", "penalty": "l2"	0.436	0.777	0.509	0.735
SGDClassifier	"loss": "perceptron", "penalty": "l1"	0.450	0.772	0.577	0.732
SGDClassifier	"loss": "squared_hinge", "penalty": "l1"	0.409	0.771	0.568	0.694
SGDClassifier	"loss": "modified_huber", "penalty": "elasticnet"	0.554	0.792	0.587	0.722
SGDClassifier	"loss": "squared_hinge", "penalty": "elasticnet"	0.554	0.783	0.617	0.715
SGDClassifier	"loss": "log_loss", "penalty": "l1"	0.479	0.775	0.542	0.737
SGDClassifier	"loss": "hinge", "penalty": "l1"	0.485	0.775	0.598	0.735
SGDClassifier	"loss": "modified_huber", "penalty": "l1"	0.363	0.767	0.645	0.736
SGDClassifier	"loss": "perceptron", "penalty": "elasticnet"	0.450	0.769	0.612	0.724
SGDClassifier	"loss": "log_loss", "penalty": "elasticnet"	0.464	0.782	0.519	0.741

Appendix C:

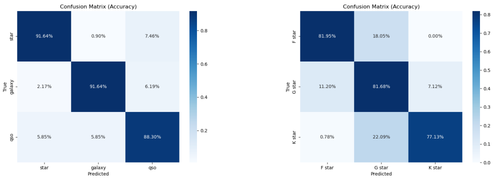

Fig. C.1: Confusion matrix of SimpleCNN1.

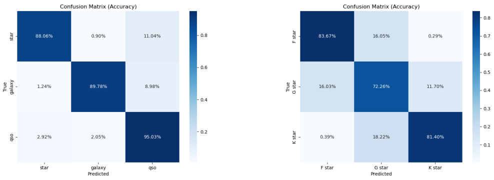

Fig. C.2: Confusion matrix of SimpleCNN2.

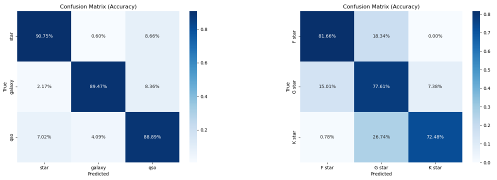

Fig. C.3: Confusion matrix of AlexNet.

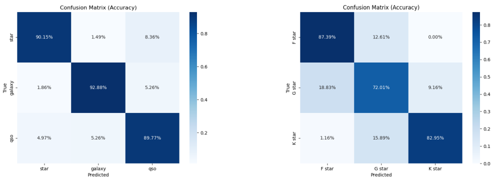

Fig. C.4: Confusion matrix of VGG16.

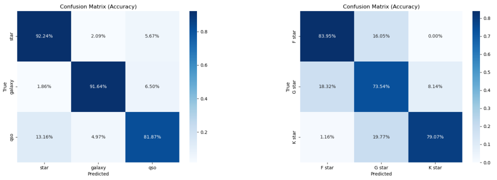

Fig. C.5: Confusion matrix of LeNet.

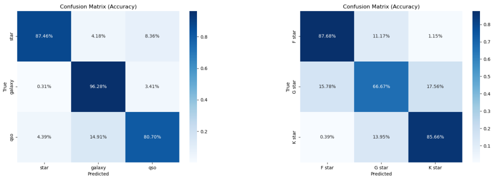

Fig. C.6: Confusion matrix of GoogLeNet.

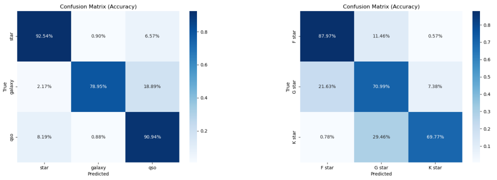

Fig. C.7: Confusion matrix of ResNet18.

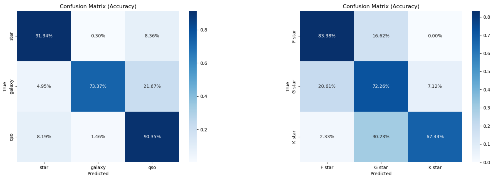

Fig. C.8: Confusion matrix of MobileNetV2.