Not getting in too deep: a practical deep learning approach to routine crystallisation image classification

Jamie Milne[1,2◑], Chen Qian[2◑], David Hargreaves[2], Yinhai Wang[2], Julie Wilson[1*].

1 Department of Mathematics, University of York, York, United Kingdom
2 AstraZeneca, Darwin Building, Cambridge Science Park, Cambridge, United Kingdom

◑These authors contributed equally to this work.
*julie.wilson@york.ac.uk

Abstract

Using a relatively small training set of ˜16 thousand images from macromolecular crystallisation experiments, we compare classification results obtained with four of the most widely-used convolutional deep-learning network architectures that can be implemented without the need for extensive computational resources. We show that the classifiers have different strengths that can be combined to provide an ensemble classifier achieving a classification accuracy comparable to that obtained by a large consortium initiative. We use eight classes to effectively rank the experimental outcomes, thereby providing detailed information that can be used with routine crystallography experiments to automatically identify crystal formation for drug discovery and pave the way for further exploration of the relationship between crystal formation and crystallisation conditions.

Author summary

X-ray crystallography can provide structural information on the molecules that play important roles in cell function and allow interactions vital for drug design to be elucidated. However, the technique requires the molecules to be crystallised and obtaining sufficiently high-quality crystals can require hundreds of experiments under different conditions. Robots have been designed to perform the microscopic experiments with imaging systems to record the results over time but automated classification of all these images is becoming essential. In this work we compare different classifiers that can be used without the need for vast computational resources and show that an ensemble classifier which combines the different strengths of four deep-learning networks is able to correctly classify the experimental results into one of eight different categories 94% of the time.

Introduction

The determination of protein structures using X-ray crystallography provides insight into the interaction between a protein and target drug compounds in the drug discovery pipeline. Despite technical advances in the crystallisation process, obtaining suitable crystals for X-ray diffraction is still a major bottleneck in drug design [1]. The current laboratory procedure still relies heavily on trial and error, although crystallisation screens have been developed to sample the search space of different conditions

intelligently. Many screens from both commercial and academic sources have been created [2], [3], [4], [5], [6], providing combinations of chemicals designed to encourage nucleation and maintain a particular pH. Such screens allow tens of thousands of unique conditions to be tested [7], with the protein construct, concentration, suitable ligand selection and experimental temperature chosen by the crystallographer.

The use of robotics now allows nanolitre volume crystallisation experiments and automated storage and imaging systems are routinely used to record the results. In high-throughput centres, thousands of experiments can be produced each day and each experiment may be imaged multiple times over several weeks. Successful crystallisation rates are thought to be less than 5% [8] and optimization experiments are usually required to obtain X-ray diffraction quality crystals from the most positive results in the initial screening. Plate storage and imaging systems can store many crystallisation plates and image the experimental results at pre-specified times whilst maintaining a constant temperature and little disturbance to the experimental plate. Each image time-series can then be extracted for examination, although the monotony of the task can often lead to erroneous classification and observer bias and inter-observer differences are frequent with the human classification of images. An investigation of agreement rates between crystallographers found these were rarely above 70% with this rate decreasing for greater numbers of classes [9]. Visual inspection of images is becoming increasingly impractical and there is an urgent need for automated image analysis.

Automated classification of protein crystallisation images was initially proposed by Zuk & Ward in 1991 when storage and imaging systems were introduced [10]. They used simple Sobel edge detection with the Hough transform to identify straight lines that could indicate the presence of crystals and suggested that more advanced algorithms could be used to monitor crystal growth and better assess the outcome of an experiment. Spraggon et. al [11](2002) expanded on this idea, using self-organising maps to classify protein crystallisation images as one of six classes, two of which corresponded to crystals, achieving a 75.6% true positive rate for crystals. Since then several research groups have reported improved results using various machine learning algorithms and different numbers of classes. For example, Cumbaa & Jurisica [12] (2005) used linear discriminant analysis to classify protein crystallisation images into four classes namely clear, precipitate, crystal and unknown, whilst Pan and coworkers [13] (2005) classified their images as containing crystals or not using support vector machines. Although this study achieved a false negative rate of less than 3%, this was at the expense of $\sim 38\%$ false positives. Following the introduction of convolutional neural networks (CNNs) for image analysis [14], Yann & Tang introduced CrystalNet, reporting a 90.8% accuracy with ten classes [15]. However, they used a carefully selected set of cleaned images, for which the class was agreed by multiple crystallographers and the same architecture has since been trained to classify more realistic protein crystallisation images with just 73.7% accuracy for ten classes [16].

The Machine Recognition of Crystallisation Outcomes (MARCO) [17] project in 2018 aimed to develop a classifier that was not specific to a particular imaging system or laboratory and used a training dataset of nearly half a million labelled images from five academic and industrial institutions, including pharmaceutical companies GlaxoSmithKline (GSK), Merck and Bristol Myers Squibb. Although classification rates of $> 94\%$ were reported for test-set images from multiple imaging platforms, problems with transferability were identified during training. A classifier trained with images from GSK, Merck and the Hauptman-Woodward Institute (HWI) achieved a classification rate of just 61.6% for a test set of images from the Collaborative Crystallisation Centre (C3, CSIRO). However, when C3 images were included in training (making up just 3% of the total training set), 87.5% of C3 images were classified correctly. In fact, transferability is still an issue with the final MARCO

classifier [18]. It achieved an accuracy of just 63% for a set of labelled crystallisation images from AstraZeneca (AZ) in our test. Unfortunately, improving the model by transfer learning with representative images from AstraZeneca is not straightforward as it was developed in collaboration with Google scientists and utilised their computational resources. The model and its weights were not published, so the only way to retrain it is to start from scratch. This has motivated the development of a new model that can be trained without the need for supercomputers. Here we compare popular deep neural network design strategies in the development of a bespoke classifier for AZ crystallisation images. The aim of this study is not simply to identify crystals but also to allow all experimental outcomes to be associated with crystallisation conditions in a way that can be exploited in further analyses.

Data and Labelling

The number of classes used to categorise crystallisation images has varied between studies, ranging from just two, *crystal* and *non-crystal*, to as many as ten [19]. If the goal is simply to identify drops containing crystals, then the former might be appropriate but, if the results are to be used in further analyses, for example, to optimise crystallisation conditions and to better understand the crystal formation process, then more classes are required. We have chosen to use the eight classes shown in Table 1 to provide information on the level of success of an experiment. Our categories include three separate crystal classes: *shootable* representing crystals of a quality that could be sent for X-ray diffraction,*optimisable* which includes very small crystals, needles and plates and *crystalline* representing microcrystalline precipitate (often interpreted as"promising looking" by crystallographers). Where initial screens do not result in any crystalline material, information on other experimental outcomes could be used to design optimisation protocols. *Phase separation*, the separation of a protein-rich phase within the crystallisation drop, can provide a boundary where crystals sometimes form later. We consider *light precipitate* and *heavy precipitate* as different outcomes, as light precipitate showing some order indicates supersaturation and can lead to crystal growth whereas heavy amorphous precipitate is unlikely to. The level of precipitate can reveal the suitability of the protein concentration, a fact that has been commercialised by Hampton Research in their Pre-Crystallisation Test (https://hamptonresearch.com). In addition to empty drops, labelled *clear*, we have a *null* class which covers the case where no drop is dispensed as well as experiments affected by factors unrelated to the experiment, such as condensation or issues with focus. Example images from each class are shown in Figure 1.

Our complete dataset consists of 25,316 images from protein crystallisation experiments, labelled according to the classes shown in Table 1. Regardless of the number of classes there will always be disagreements on classification by humans with classes that are not discrete. After a training session led by an experienced crystallographer, a subset of 3095 images was labelled independently by three researchers, allowing the level of agreement to be assessed. Of these images, 2745 were unique with an additional set of 50 duplicated 7 times interspersed. Images were labelled in the same order by each classifier and intra-observer bias measured by the number of times that the 50 repeated images were given the same label. At least 86% of the images were given the same label four out of seven times, but scores for more than four agreements dropped drastically with scores for an individual giving the same label all seven times between 10% and 28%. Overall agreement rates between different human classifiers ranged from 50% to 70%, showing the difficulty in classification with continuous outcomes.

Given the low agreement rates between classifiers, a set of 5972 images were given

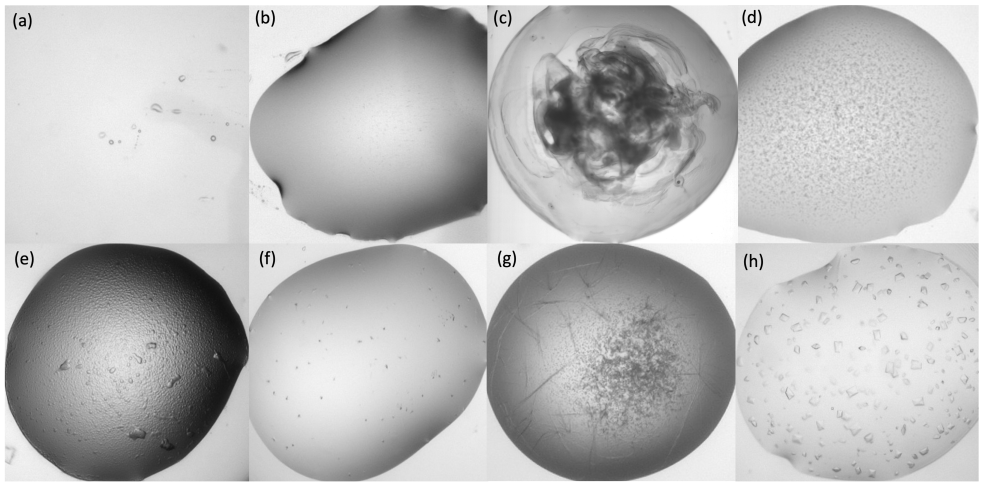

Fig 1. Example images from each class. (a) shows a *null* image in which no drop has been dispensed;(b) shows a *clear* drop;(c) shows *heavy precipitate* where the protein is denatured;(d) shows *light precipitate*; (e) shows *phase separation*; (f) shows *crystalline* material; (g) shows *optimisable* crystals and (h) shows *shootable* crystals.

labels agreed by three scientists together to produce an initial training set. The model produced with this training data was then used to classify further images and a simple graphical user interface developed to check the results (see Supplementary Fig. S2. Any image for which the classification was considered correct was added to the training set with the assigned label, whereas any image considered misclassified was relabelled before being added. This allowed the size of the training set to increase rapidly but reliably in an iterative procedure that continuously improved the classifier.

As each experiment is imaged multiple times, often with little change between consecutive images, only images from days 1, 8 and 13 were ever included in the training set to reduce redundancy.

Images are mainly from two particular projects, with 7,395 images associated with the receptor tyrosine kinases EGFR [20] and c-KIT [21]. A further 8,922 images are from initial crystallisation screens involving several different proteins to provide the model with a greater variety of experimental outcomes and different crystal forms. In total, 16,317 images were used for training the classifier and Table 1 shows the number in each class.

class number	**class name**	**before augmentation**	**after augmentation**
1	null	636	4452
2	clear	3589	3589
3	heavy precipitate	3493	3493
4	light precipitate	4520	4520
5	phase separation	668	4676
6	crystalline	1834	3668
7	optimisable	652	4564
8	shootable	925	4625

Table 1. Training Data. The eight categories used to classify images in our dataset and the number of images in the training set associated with each class, before and after data augmentation.

Classification results were validated with three independent test sets. To assess the effect of including very similar images in the training and test sets, our first test set consists of images from time courses represented in the training data but from different batches (i.e. from days other than 1, 8 and 13). The second test consists of images related to proteins represented in the training set but from different experimental plates and the third test set comprises images from independent experiments associated with the chemotherapy drug Trastuzumab [22], for which no images were included in the training set. Table 2 summarises the three test sets with the number of images in each.

	number of images in each class		**similarity to training set**
Test Set 1 (N = 3000)	Shootable	131	Different time point of represented experiment
	Optimisable	296	
	Crystalline	379	
	Phase separation	187	
	Light precipitate	435	
	Heavy precipitate	859	
	Clear	418	
	Null	295	
Test Set 2 (N = 3000)	Shootable	107	Different experiment from represented project
	Optimisable	135	
	Crystalline	267	
	Phase separation	295	
	Light precipitate	452	
	Heavy precipitate	1054	
	Clear	529	
	Null	161	
Test Set 3 (N = 2999)	Shootable	18	Unrepresented project
	Optimisable	27	
	Crystalline	91	
	Phase separation	364	
	Light precipitate	598	
	Heavy precipitate	1275	
	Clear	614	
	Null	12	

Table 2. Test data sets. The number of images in each class for the three different test sets with their similarity to images in the training set.

As colour images multiply the complexity of models and increase computational cost, our RGB images are converted to grayscale, or more specifically luminant (L), where $L = 0.299R + 0.587G + 0.114B$. Each image is cropped to 800x800 pixels, covering the protein crystallisation drop. Class imbalance is a fundamental issue in image classification as unbalanced training data results in models that are biased towards the better represented classes. With an estimated crystallisation rate of $< 5\%$, it is especially problematic for automated crystallisation image analysis. Although we limited the number of images in other classes, the three crystal classes (*crystalline*, *optimisable* and *shootable*) account for just 20.9% of the training data with just 5.7% labelled *shootable*. As recognising crystals is a key aim, we incorporated data augmentation into the pre-processing stage to balance the number of images in each training classes. Multiple independent transformations are applied to images from underrepresented classes, including random rotations, horizontal and vertical flips, and contrast, brightness and resolution adjustments. Each class is duplicated by the

rounded inverse of its proportion to the largest class *light precipitate*. The training set is summarised in Table 1.

Model Comparison

The MARCO classifier [23] used an inception-V3 architecture with an additional convolutional layer that reduces the image dimensions from 599x599 to 299x299 pixels. Inception-V3 is a variation on the original Inception module in which a set of convolutional operations occur in parallel before concatenation on output. The V3 module incorporates larger convolutions and regularisation in the form of batch normalisation and label smoothing. This introduces greater complexity but reduces the top-5 error on the ImageNet dataset from 6.67% to 4.2% in comparison to Inception-V1 (GoogLeNet [24]).

Advances in image classification are constantly being made, leading to reduced training times and computational cost as well as improved classification rates. The building blocks, or layers, used to create the architecture of a deep learning network vary and are often combined with new architectures to extend and improve previous ideas. The application programming interface (API) Keras [25] compared network architectures using the ImageNet validation set [26] and provided the results in terms of Top-1 accuracy, where the class given the highest probability is correct, and Top-5 accuracy, which means that one of the five highest probability answers is the correct class. The number of parameters involved and training times are also reported. Computational efficiency in relation to accuracy has been investigated previously for the most popular CNNs [27] and, although NASNet-A-Large was found to have the greatest Top-1 and Top-5 accuracies, it also had the greatest computational cost. Here. we chose to investigate the four architectures with the best-reported Top-1 and Top-5 accuracy that did not exceed our GPU capacity, these being:

1. ResNet50 [28]
2. DenseNet121 [29]
3. InceptionV3 [23]
4. Xception [30]

ResNet provides a popular architecture with shortcuts that allow skipped connections to stabilise training but this can compromise the learning capacity of the network. This is mitigated by DenseNet by concatenating all previous feature maps rather than summing them as in ResNet and results in a 2.4% improvement in Top-1 accuracy on the ImageNet validation data with 5.4 million fewer parameters. Similarly, Xception, *aka* 'extreme Inception' was designed to improve on InceptionV3 and achieved a 1.1% increase in accuracy on the ImageNet images while using a million fewer parameters.

Network parameters were optimised using DenseNet121 architecture with a subset of 8,000 images from the MARCO project, split into training (80%) and validation (20%) sets. During optimisation, training was carried out for 20 epochs using a cross-entropy loss function and improvements in validation accuracy were identified as individual parameters were changed. Tests indicated that two additional fully connected layers, separated by dropout layers reduced overfitting and increased validation accuracy. The network parameters found to be optimal were used to train new classifiers with the four chosen architectures and the training data shown in table 1. A consistent batch size of 16 was used for all architectures regardless of complexity, whilst training occurred for 100 epochs in each case with no early stopping. Weights were optimised using stochastic

gradient descent with the Adam optimiser and an initial learning rate of $2e^{-4}$. The learning rate was divided by 2 if the loss had not decreased in 5 epochs.

Transfer learning allows the weights from pre-trained models to be re-used as a starting model for new deep learning problems to save time. However, the models provided for the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), are based on RGB images and therefore, in order to adapt them to work with our grayscale images, we created two Keras Application models (keras.io/api/applications) for each framework, one accepting RGB images using the ImageNet weights and the other accepting gray scale images with no initial weights specified. The ImageNet weights were transferred to the gray scale model from the layer at which the input shape allowed this. This approach has been shown to improve the efficiency of the model by reducing the number of initial channels without significantly affecting precision [31]. All training was implemented using TensorFlow and Keras and run on four NVIDIA Tesla V100 graphic processing units. The training times per epoch are shown in Table 3.

classifier	**hours:minutes:seconds**
DenseNet121	0:23:08
ResNet50	0:22:52
InceptionV3	0:18:45
Xception	1:16:24

Table 3. Training times per epoch for each classifier.

Results

Fig 2 shows the performance of the four classifiers in ROC space when considering the three classes, *crystalline*, *optimisable* and *shootable* together as positive results and all other classes as negatives. As expected, the results for all architectures are best for test set 1, for which images from the same time series were included in the training data. The ResNet50 classifier stands out as most different with noticeably higher sensitivity for Test sets 1 and 2 at the expense of much lower specificity. There are fewer false positives for Test Set 3, comprising images from experiments on a protein not represented in the training data, with this classifier but the sensitivity is also much lower. The DenseNet121 classifier has similar sensitivity for Tests sets 2 and 3 but with lower specificity for Test set 2, where images from experiments with the same protein but not the same time series were included in the training set. Xception gives very similar results to DenseNet for Test set 3, whereas Inception-V3 produces fewer true positives. However, for Test set 2, Inception-V3 shows a slight improvement over DenseNet, whereas a reduction in false positives is outweighed by a reduction in true positives with Xception. It should be noted that the number of images in each class differs between test sets and that test set 3 in particular has few positive results (see Table 2).

Accuracy is measured as the proportion of classifications that are correct irrespective of class and can therefore be misleading when classes are unbalanced. Cohen's Kappa, originally devised to compare interrater agreement, allows for class imbalance when used to assess classification results and so can provide a better representation of performance. Fig 3 shows the Kappa and accuracy scores for the three test sets along with two other common performance metrics, precision and the F1-score. Although the latter two scores are typically used for binary classification, one-vs-all scores can be computed for each class and the weighted average used to provide an overall score. Regardless of the performance metric, it can be seen that DenseNet121 outperformed the other classifiers on all three test sets. Xception produced the next best results,

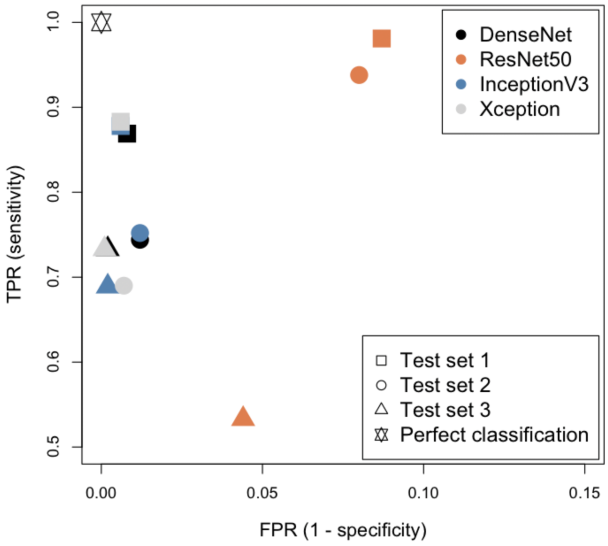

Fig 2. ROC space comparison of the four chosen architectures with the three different test data sets in Table 2. Note that only the pertinent area of ROC space is shown.

followed by InceptionV3 and ResNet50 with little to choose between them for Test sets 2 and 3. Results are also shown in Supplementary Table table:differentmetrics.

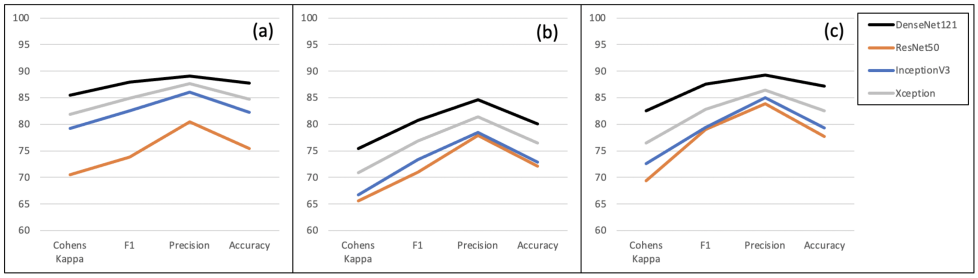

Fig 3. Performance metrics for the four chosen architectures on (a) Test set 1, (b) Test set 2 and (c) Test set 3.

It is somewhat surprising that the results obtained for Test set 3, using images from a project not represented in the training data, are consistently better than those for Test set 2, for which images from experiments with same proteins are included. These results are not explained by the fact that there are far fewer examples in the crystal classes. In fact, the results for Test set 3 are proportionally similar to those for Test Set 1 for most classes as can be seen from the full confusion matrices in Supplementary Information (Figs S3 - S6).

Fig 4 compares the results between test sets obtained with the DenseNet121 classifier. Here balanced accuracy, i.e the average of sensitivity and specificity, is plotted for Test set 1 against each of the other two test sets. For five of the classes, Test set 3 gives very similar results to Test set 1, as can be seen from their proximity to the dotted diagonal line, whereas the results for Test set 2 are lower. Here *heavy precipitate* is not only confused with *light precipitate*, but also *crystalline*. There is also more confusion of *crystalline* with *phase separation* for Test set 2. Test sets 2 and 3 results are more similar for the other three classes (*light precipitate, optimisable* and *null*) although, apart from *null* which also has few examples in Test set 3, results for Test set 3 are still closer to those for Test set 1.

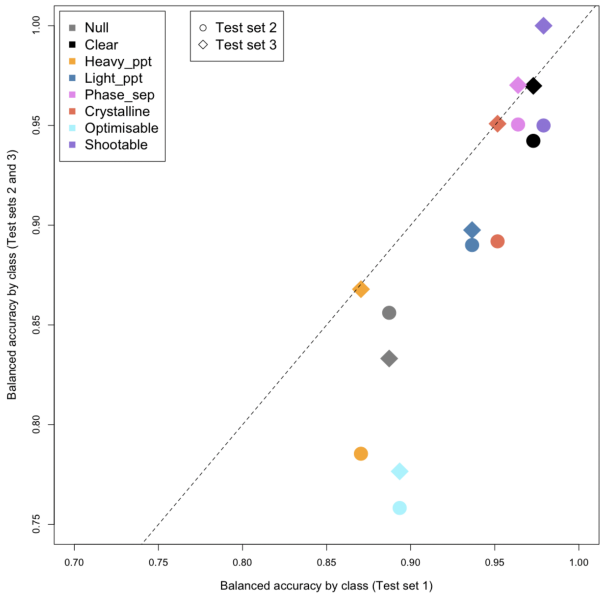

Fig 4. Comparison of Test set results by class. For each class, the balanced accuracy obtained for Test Set 1 is plotted against the balanced accuracy for Test Sets 2 and 3, identified by different symbols. The results shown were obtained using the DenseNet121 classifier.

The lowest class accuracy overall is for phase separation with Xception, being confused with both precipitate and crystals. The Xception classifier also confuses both null and clear images with precipitate as does the InceptionV3 classifier. Other regular errors with InceptionV3 are crystals classified as precipitate and phase separation classified as crystals. With ResNet50 the lowest accuracies are for phase separation most often confused with crystal classes and crystals confused with precipitate. As seen earlier, the best results overall are obtained with the DenseNet121 classifier with the lowest accuracy for null images, mainly confused with clear although $\sim 10\%$ of precipitate images are classified as crystals.

classifier	null	clear	heavy ppt	light ppt	phase sep	crystalline	optimisable	shootable
DenseNet121	0.97	0.95	0.93	0.93	0.86	0.79	0.92	0.94
ResNet50	0.86	0.91	0.87	0.88	0.82	0.89	0.71	0.99
InceptionV3	0.93	0.98	0.77	0.87	0.91	0.85	0.96	0.84
Xception	0.96	0.97	0.81	0.89	0.97	0.81	0.97	0.88
Ensemble4	0.96	0.99	0.97	0.98	0.99	0.98	0.90	1.00
Ensemble3	0.96	0.98	0.94	0.95	0.98	0.95	0.91	97.0

Table 4. Balanced class accuracies for Tests sets 2 and 3 combined obtained using ensemble classification in comparison to those obtained for individual classifiers. Ensemble4 combines the results of all four classifiers, whereas emsemble3 does not use the predictions from the Xception classifier. Accuracies greater than 90% are highlighted.

The differences in misclassifications between classifiers suggest the possibility of

DenseNet 121	Predicted class				
True class	null	clear	ppt	phase-sep	crystal
null	75.4	11.2	4.6	2.9	5.9
clear	0.1	92.6	0.6	0.5	6.1
ppt	0.3	0.8	84.9	4.2	9.8
phase-sep	0.0	0.0	1.0	94.6	4.4
crystal	0.1	0.1	1.7	3.4	94.8

ResNet 50	Predicted class				
True class	null	clear	ppt	phase-sep	crystal
null	86.8	2.6	3.9	0.9	5.7
clear	7.7	90.9	0.2	0.4	0.7
ppt	2.2	3.8	86.3	4.7	3.0
phase-sep	0.8	2.5	2.5	79.5	14.7
crystal	1.0	3.0	11.0	4.9	80.1

Inception V3	Predicted class				
True class	null	clear	ppt	phase-sep	crystal
null	87.6	0.2	11.8	0.2	0.7
clear	5.1	76.8	12.6	2.0	3.6
ppt	0.4	0.1	95.8	1.0	2.8
phase-sep	0.6	0.0	5.8	83.8	9.8
crystal	0.1	0.1	9.7	2.0	88.1

Xception	Predicted class				
True class	null	clear	ppt	phase-sep	crystal
null	82.2	3.5	11.4	0.0	2.9
clear	1.8	84.0	11.3	0.3	2.6
ppt	0.3	0.1	93.7	0.2	5.8
phase-sep	0.2	0.2	13.1	65.8	20.8
crystal	0.3	0.4	3.5	0.2	95.6

Fig 5. Confusion matrices showing the results obtained with the four different classifiers for Test sets 2 and 3 combined .The number of classes has been reduced for easier comparison by combining *light precipitate* and *heavy precipitate* as just *precipitate* and the three classes *crystalline, optimisable* and *shootable* as *crystal*.

classifier	**Accuracy**	**Cohen's Kappa**
DenseNet121	0.83	0.81
ResNet50	0.74	0.68
InceptionV3	0.78	0.73
Xception	0.81	0.88
Ensemble4	0.95	0.94
Ensemble3	0.92	0.90

Table 5. Overall accuracy and Kappa values for ensemble classification in comparison to those obtained from individual classifiers. Ensemble4 combines the results of all four classifiers, whereas emsemble3 does not use the predictions from the Xception classifier.

improved classification from ensemble classification. Indeed, majority voting with all four classifiers yields improved balanced accuracies over the DenseNet121 classifier of 13% and 19% for phase separation and crystalline examples respectively with a reduction of just 1 or 2% for *null* and *optimisable* (Table 4). Table 5 shows improvements of up to 21% for overall accuracy and 26% for Cohen's Kappa when the predictions of all four classifiers are combined (Ensemble4) with improvements of 12% for accuracy and 13% for Kappa over the DenseNet121 classifier. We found that choosing the class randomly in the case of tied votes actually gave slightly worse results than simply taking the class that comes up first and realised that this was due to the order that the predictions were provided. The predictions from the DenseNet121 classifier, shown to be the best individual classifier for our data, are given first and so are given priority in the case of ties. Further investigation showed that changing the order of the other three classifiers did not make any difference to the results. However, as the Xception architecture required so much extra training time, we also tried combining the results without this classifier (Ensemble3). Tables 4) and 5 show that

correct classifications fall by 3-4%.

Discussion

The MARCO classifier was trained with over 440 thousand images from five different academic and industrial sources, achieving an accuracy of $\sim 94\%$ on test images from these institutions [17]. However, even when the same imaging system is used, subtle differences in the set-up result in much lower accuracies for images from other organisations. Although the classifier is freely available [18] the weights are not, which means that tuning with additional images is not possible and the only way to incorporate new information is to retrain from scratch. Rosa and coworkers characterised the image data used in the original MARCO model investigated the training settings most likely to enhance the local performance of a classification model based on these images [32]. However, we found that using only our own images gave better results than including even subsets of the MARCO images. Our training set is very small in comparison but diverse and carefully labelled and we have shown that similar results are achieved for independent test data whether from projects represented in the training data or not.

We compared four of the most widely-used network architectures, chosen by their performance reported by the Keras API but taking into account computational limitations. It is possible that with greater computational capacity more advanced models, such as inception-v4 and SENet-154, could further improve classification rates as they do with the ImageNet validation data. Each of our chosen architectures has been used in recent studies on the detection of COVID-19 from chest X-Ray images, sometimes in combination. Rahimzadeh and Attar concatenated the output features from Xception and ResNet50V2 to provide a classification model that could differentiate between COVID-19 and pneumonia [33], achieving an overall accuracy of 91.4%. With many more normal and pneumonia cases, they had just 30 COVID-19 cases in the test set for which the sensitivity was 75.3%. A balanced dataset comprising 320 chest X-Ray images from each of COVID-19 patients and healthy volunteers was used in the ratio 50:20:30 for training, validation and testing [34]. Zhang and colleagues compared the results of three DenseNet architectures (121,169 and 201) with different transfer learning settings. They found that DenseNet201 gave the best results with an overall accuracy of 96.9% and suggested that its deeper neural structure could learn more complicated patterns. A comparison of three other architectures (ResNet50, InceptionV3 and VGG16) was carried out by Guefrechi and colleagues with a larger data set of chest X-Ray images [35]. The full data set consisted of 623 COVID-19 positive images augmented to 2000 together with 3000 COVID-19 negative images. They fine-tuned each classifier and obtained similar results for the different architectures, with accuracies of 97.1% to 98.3% on test data.

Previous benchmark analysis of different architectures found that Xception gave the highest accuracy, followed by InceptionV3, ResNet50 and then DenseNet121 [27], whereas, for our data, the DenseNet121 architecture outperformed the other architectures. However, we did obtain very similar accuracy with Xception and, in fact, did give a higher Kappa score for Test sets 2 and 3 combined. Although the ResNet50 architecture produced the worst predictions overall, this classifier had a high crystal detection rate of 93.8%. It does however produce more false positives as can be seen in Figure 2. The training time for the Xception classifier was four times that for InceptionV3, requiring 127 hours over the 100 epochs in comparison to between 31 and 39 hours for the other three classifiers. As the four classifiers had different strengths in terms of particular class accuracies, we were able to combine their results in an ensemble classifier using majority voting and achieve accuracy comparable to that seen

in the MARCO project. We consider the increase in accuracy over the best individual classifier to be worth the combined training time.

We have chosen to categorise the results of crystallisation experiments using eight classes rather than the four used in the MARCO projects in order to provide useful information for further analysis. Whereas the MARCO classes combined both good and bad results in the ambiguously named *other* class, we aim to show just how good an outcome is by having more classes. With the exception of *null*, which could represent unavoidable technical problem rather than an experimental outcome, our eight classes reflect some ranking, even within the three crystal classes, that could inform further work. We have developed a graphical user interface (GUI) to visualise the results for each experimental plate and Fig. 6 shows an example obtained using the Morpheus crystallisation screen [4]. This screen was designed as a grid with columns covering 4 different precipitant-cryoprotectant mixes for three different buffer systems (pH 6.5, 7.5 and 8.5) and rows corresponding to different additives. Patterns can be seen in Fig. 6 with certain columns being associated with particular classes, while some additives are noticeable for being different, for example row B (halides) and row G (carboxylic acids). As with the MARCO Polo interface, developed to display and share the results of the MARCO classifier [36], our GUI allows images in the time course to be displayed and provides experimental details. Patterns such as those seen in Fig. 6 suggest that successful conditions could potentially be related to particular protein properties, allowing optimal conditions for new proteins to be predicted.

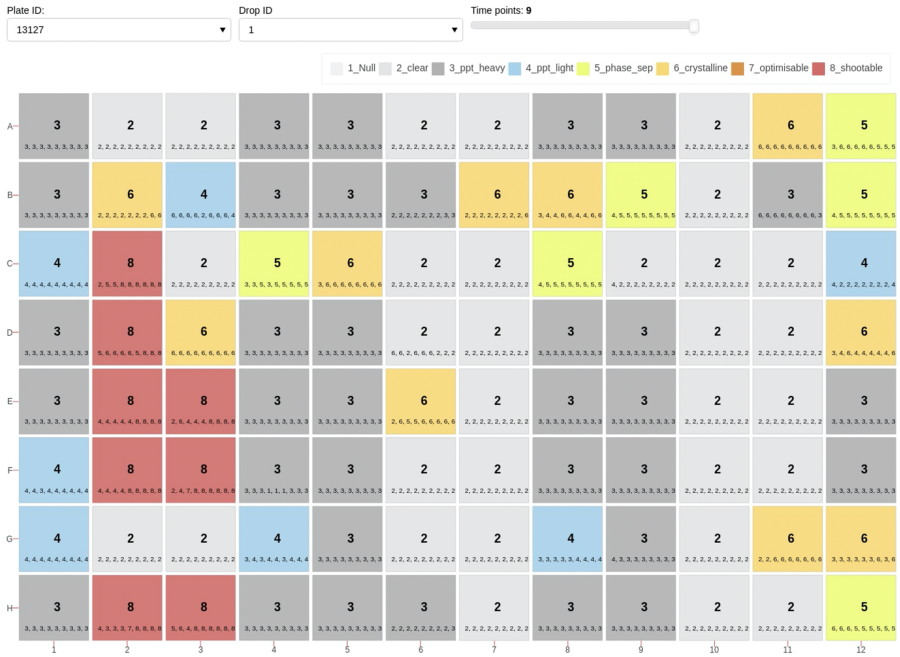

Fig 6. Visualisation of experimental outcomes obtained using the Morpheus crystallisation screen [4]. In addition to being labelled with the predicted class number, each well in the plate is colour-coded according to the predicted class. The sequence of numbers in small text for each well shows the class numbers predicted earlier in the time series.

Conclusion

We have shown that a bespoke ensemble classifier can be trained to classify images from crystallisation experiments without the need for supercomputers or a huge training data set. Comparison of four popular deep learning architectures revealed interesting differences in class accuracies between these classifiers. While DenseNet121 provided the best overall accuracy, each architecture was optimal for at least one particular class. The results in Table 4 show that DenseNet121 actually only gave the best results for *null* images and the two precipitate classes, whereas InceptionV3 gave the highest accuracy for it clear images, Xception was best for *phase separation* and *optimisable* images and ResNet50 produced the best results for images in the *crystalline* and *shootable* classes. Although the overall accuracy achieved with ResNet50 is 9% lower than that for DenseNet121, this classifier did in fact have an accuracy for *crystalline* images 10% higher than that with the DenseNet architecture. These differences in missclassification suggested ensemble classification could improve the results and the use of majority voting with all four classifiers increased the overall accuracy from 83%, obtained for the best individual classifier, to 95% and Cohen's Kappa from 81% to 94%.

Acknowledgements

This work was possible thanks to the award of an EPSRC CASE studentship to JM (reference EP/V519807/1). This was co-funded by AstraZeneca UK Ltd as industrial CASE partner. YW, CQ and DH are employed by AstraZeneca UK Ltd and support was also provided in the form of salaries for these authors. We would also like to thank scientists at AstraZeneca UK Ltd for providing data and feedback.

References

1. Rathore I, Mishra V, Bhaumik P. Advancements in macromolecular crystallography: From past to present. Emerging Topics in Life Sciences. 2021;5(1):127–149.

2. Fazio VJ, Peat TS, Newman J. A drunken search in crystallization space. Acta Crystallographica Section F: Structural Biology Communications. 2014;70(10):1303–1311.

3. Abrahams G, Newman J. Data and Diversity Driven Development of a Shotgun Crystallisation Screen using the Protein Data Bank. bioRxiv. 2021;.

4. Gorrec F. The MORPHEUS protein crystallization screen. Journal of applied crystallography. 2009;42(6):1035–1042.

5. Newman J, Egan D, Walter TS, Meged R, Berry I, Ben Jelloul M, et al. Towards rationalization of crystallization screening for small-to medium-sized academic laboratories: the PACT/JCSG+ strategy. Acta Crystallographica Section D: Biological Crystallography. 2005;61(10):1426–1431.

6. Brzozowski AM, Walton J. Clear strategy screens for macromolecular crystallization. Journal of applied crystallography. 2001;34(2):97–101.

7. Abrahams GJ, Newman J. BLASTing away preconceptions in crystallization trials. Acta Crystallographica Section F: Structural Biology Communications. 2019;75(3):184–192.

8. Schmit JD, Dill K. Growth rates of protein crystals. Journal of the American Chemical Society. 2012;134(9):3934–3937.

9. Wilson J. Automated classification of images from crystallisation experiments. In: Industrial Conference on Data Mining. Springer; 2006. p. 459–473.

10. Zuk WM, Ward KB. Methods of analysis of protein crystal images. Journal of crystal growth. 1991;110(1-2):148–155.

11. Spraggon G, Lesley SA, Kreusch A, Priestle JP. Computational analysis of crystallization trials. Acta Crystallographica Section D: Biological Crystallography. 2002;58(11):1915–1923.

12. Cumbaa C, Jurisica I. Automatic classification and pattern discovery in high-throughput protein crystallization trials. Journal of structural and functional genomics. 2005;6(2):195–202.

13. Pan S, Shavit G, Penas-Centeno M, Xu DH, Shapiro L, Ladner R, et al. Automated classification of protein crystallization images using support vector machines with scale-invariant texture and Gabor features. Acta Crystallographica Section D: Biological Crystallography. 2006;62(3):271–279.

14. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems. 2012;25.

15. Yann M, Tang Y. Learning deep convolutional neural networks for x-ray protein crystallization image analysis. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 30; 2016. p. 1373–1379.

16. Ghafurian S, Orth P, Strickland C, Su H, Patel S, Soisson S, et al. Classification of protein crystallization x-ray images using major convolutional neural network architectures. arXiv preprint arXiv:180504563. 2018;.

17. Bruno AE, Charbonneau P, Newman J, Snell EH, So DR, Vanhoucke V, et al. Classification of crystallization outcomes using deep convolutional neural networks. PLOS one. 2018;13(6):e0198883.

18. GitHub Marco classifier;. `https://github.com/tensorflow/models/tree/master/research/marco`.

19. Saitoh K, Kawabata K, Asama H, Mishima T, Sugahara M, Miyano M. Evaluation of protein crystallization states based on texture information derived from greyscale images. Acta Crystallographica Section D: Biological Crystallography. 2005;61(7):873–880.

20. Carpenter G, Lembach KJ, Morrison M, Cohen S. Characterization of the binding of 125-I-labeled epidermal growth factor to human fibroblasts. Journal of Biological Chemistry. 1975;250(11):4297–4304.

21. Lennartsson J, Rönnstrand L. Stem cell factor receptor/c-Kit: from basic science to clinical implications. Physiological reviews. 2012;92(4):1619–1649.

22. Valabrega G, Montemurro F, Aglietta M. Trastuzumab: mechanism of action, resistance and future perspectives in HER2-overexpressing breast cancer. Annals of oncology. 2007;18(6):977–984.

23. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2016. p. 2818–2826.

24. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2015. p. 1–9.

25. Keras API;. `https://keras.io/`.

26. ImageNet data;. `https://image-net.org/`.

27. Bianco S, Cadene R, Celona L, Napoletano P. Benchmark analysis of representative deep neural network architectures. IEEE access. 2018;6:64270–64277.

28. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2016. p. 770–778.

29. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017. p. 4700–4708.

30. Chollet F. Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017. p. 1251–1258.

31. Wightman R. How can I use a pre-trained neural network with grayscale images?; 2022 Accessed: 2022-8-01. Available from: `https://stackoverflow.com/a/54777347`.

32. Rosa N, Watkins CJ, Newman J. Moving Beyond MARCO. bioRxiv. 2022;.

33. Rahimzadeh M, Attar A. A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2. Informatics in medicine unlocked. 2020;19:100360.

34. Zhang YD, Satapathy SC, Zhang X, Wang SH. Covid-19 diagnosis via DenseNet and optimization of transfer learning setting. Cognitive computation. 2021; p. 1–17.

35. Guefrechi S, Jabra MB, Ammar A, Koubaa A, Hamam H. Deep learning based detection of COVID-19 from chest X-ray images. Multimedia Tools and Applications. 2021;80(21):31803–31820.

36. Holleman ET, Duguid E, Keefe LJ, Bowman SE. Polo: an open-source graphical user interface for crystallization screening. Journal of applied crystallography. 2021;54(2):673–679.