Journal of Pharmaceutical Sciences 107 (2018) 999-1008

Contents lists available at ScienceDirect

Journal of Pharmaceutical Sciences

journal homepage: www.jpharmsci.org

Pharmaceutical Biotechnology

Deep Convolutional Neural Network Analysis of Flow Imaging Microscopy Data to Classify Subvisible Particles in Protein Formulations

Christopher P. Calderon [1, 2, *], Austin L. Daniels [2], Theodore W. Randolph [2, *]

[1] Ursa Analytics, Inc., Denver, Colorado 80212-1962
[2] Department of Chemical and Biological Engineering, University of Colorado, Boulder, Colorado 80309-0596

ARTICLE INFO

Article history:
Received 2 October 2017
Revised 27 November 2017
Accepted 7 December 2017
Available online 18 December 2017

Keywords:
protein aggregation
image analysis
quality control
regulatory science
protein formulation

ABSTRACT

Flow-imaging microscopy (FIM) is commonly used to characterize subvisible particles in therapeutic protein formulations. Although pharmaceutical companies often collect large repositories of FIM images of protein therapeutic products, current state-of-the-art methods for analyzing these images rely on low-dimensional lists of "morphological features" to characterize particles that ignore much of the information encoded in the existing image databases. Deep convolutional neural networks (sometimes referred to as "CNNs or ConvNets") have demonstrated the ability to extract predictive information from raw macroscopic image data without requiring the selection or specification of "morphological features" in a variety of tasks. However, the inherent heterogeneity of protein therapeutics and optical phenomena associated with subvisible FIM particle measurements introduces new challenges regarding the application of ConvNets to FIM image analysis. We demonstrate a supervised learning technique leveraging ConvNets to extract information from raw images in order to predict the process conditions or stress states (freeze-thawing, mechanical shaking, etc.) that produced a variety of different protein particles. We demonstrate that our new classifier, in combination with a "data pooling" strategy, can nearly perfectly differentiate between protein formulations in a variety of scenarios of relevance to protein therapeutics quality control and process monitoring using as few as 20 particles imaged via FIM.

Introduction

Particulate matter in therapeutic protein products is the focus of increased attention due both to industrial quality control and patient safety concerns.[1-5] "Subvisible" particles (defined here as objects ≤25 μm in size) are contained in all commercial therapeutic prote in formulations.[3,6,7] Subvisible particles may be composed of aggregated proteins or nonbiological materials (e.g., silicone oil).[3,8-11] Particles of size less than 10 μm are not (currently) explicitly regulated by the U.S. Food and Drug Administration,[12] but they can account for as much as 90% of the particulate matter in a therapeutic protein product[1] and have been associated with a variety of immune responses.[4,6,13,14] However, not all particles are immunogenic or harmful[15]; the precise structure-immunogenicity relationships associated with particles that induce immunogenic responses are currently unknown.[16,17] Determination of the particle characteristics that are associated with risk to patients is complicated by the fact that protein aggregates and other particles exhibit a high degree of heterogeneity in size, shape, and composition.[18-20]

Flow-imaging microscopy (FIM) is a powerful tool that is capable of recording complex images of single subvisible particles.[4,8,21] In FIM experiments, a small liquid sample is pumped through a microfluidic flow cell, and a digital microscope is used to record 10^3-10^6 images of individual particles in a single experiment. The images of subvisible particles returned by FIM are believed to contain a significant amount of structural information about the particles in a given sample.[5] Characterization techniques capable of leveraging the structural information embedded in FIM images show promise as tools for evaluating therapeutic protein drugs at different stages of their life spans (from the manufacturing plant to delivery to patients) and early steps in this direction have been recently proposed.[5]

Historically in FIM analysis, accurate characterization of subvisible particles from FIM images has required explicit identification of the relevant morphological features of the particles from the

This article contains supplementary material available from the authors by request or via the Internet at https://doi.org/10.1016/j.xphs.2017.12.008.

* *Correspondence to:* Christopher P. Calderon (Telephone/Fax: +1-720-663-9923) Theodore W. Randolph (Telephone/Fax: +1-303-492-4776).
E-mail addresses: chris.calderon@ursaanalytics.com (C.P. Calderon), Theodore.Randolph@Colorado.edu (T.W. Randolph).

https://doi.org/10.1016/j.xphs.2017.12.008
0022-3549/

raw FIM images. However, manual identification of the relevant features of particles for characterization tasks from the raw FIM images is challenging. Although human subject-matter experts can often observe morphological features of the particles that vary from image to image and data set to data set, it is difficult and time-consuming to express these observations in terms of changes in structural features that can be used for characterization. This inability to identify the relevant features of these subvisible particles has hindered previous efforts to characterize these particles.

Most FIM image analysis methods developed to date only use the small number of morphological features reported by FIM instruments such as aspect ratio, compactness, and particle intensity to build a "representation" of the image—a numerical description of the features of the image that can be used for characterization and classification tasks.[3,5,21,22] However, this short list of features neglects a significant amount of potentially relevant structural information contained in the full RGB (or grayscale) FIM images. Furthermore, the short list of morphologic features contains features that are highly correlated, further reducing their information content.[3] It would be desirable to harness the large amount of complex digital information encoded in images and to automatically extract the relevant features for a given classification task without requiring the selection or manual specification of "morphologic features."

One candidate solution to the shortcomings of current FIM image analysis methods is the application of deep convolutional neural networks (CNNs or "ConvNets") along with supervised classification.[23] The use of ConvNets for analyzing macroscopic images has exploded in the recent years.[23] ConvNets are now capable of matching or exceeding expert human performance in a variety of supervised learning applications.[23] By exploiting the power of supervised learning and large image data repositories, ConvNets are already revolutionizing many real-world applications that rely on accurate image analysis.[23-26] Recent improvements in algorithms, advances in graphical processor unit computing technology, and (perhaps most importantly) a deluge of digital data in almost every application domain[23,27-33] have improved our ability to automatically classify images using ConvNets, and in many cases perform such classification with human-level precision.[23] These advances enable "deep" ConvNets consisting of multiple hidden layers to be efficiently and robustly estimated. Deep ConvNets are attractive as they leverage large volumes of data to learn the best representation of images for a given classification task,[23] circumventing the need to specify lists of "morphologic properties" that best represent the particle as is done in current approaches. Data-driven representation learning is the key to match or exceed human performance,[23] but carefully designing deep ConvNets to reliably accommodate the statistical nuances of a given application benefits from the close interaction between computational scientists, statisticians, and subject-matter experts.[25]

Supervised learning methods using deep ConvNets rely on labeled data sets for training. For instance, a ConvNet model designed to classify pictures of animals might be trained with sets of pictures labeled according to their respective classes, for example, sets of animal pictures that have been labeled as depicting elephants, cats, or dogs. In recent years, ConvNets have been the most successful in supervised classification, that is, classification tasks where accurately labeled image sets are available on which to train the network. For example, a recently developed model that allows automated diagnoses of skin cancer based on digital images was trained using a collection of 129,450 clinical images (representing more than 2000 skin diseases) that had been labeled by dermatogists.[25] However, one key challenge in training deep ConvNet models to classify FIM images of particles within therapeutic protein formulations is the lack of appropriately labeled data sets wherein each particle image can be assigned a class (e.g., the aggregation mechanism that generated a given particle) that we want to train a ConvNet to identify. Although it would be useful if FIM images of subvisible particles in protein formulations could be labeled at the "particle level of detail" (e.g., proteinaceous vs. nonproteinaceous particles, or even finer levels of detail such as "this is a protein aggregate induced by freeze-thawing stress" or "this is a particle associated with increased risk of adverse immune response"), unfortunately such labeled data sets are difficult to obtain. Human subject-matter experts presented with images of single particles from an FIM analysis of particles within a therapeutic protein formulation would have trouble assigning the correct label to single FIM images, even in situations where a small fixed number of descriptive class labels are provided a priori, for example, see Figure 1. In other words, the error rate of the "optimal" classifier[34] based on a single FIM image is far from 0 (and also unknown) in almost all FIM image classification tasks. This is in contrast to problems of macroscopic image classification, where one is given a simple list such as "dog, cat, or elephant," and the goal is to determine which of the given preset labels applies to an image, a task for which both most humans and modern deep ConvNets (trained via supervised learning) exhibit almost a 0 classification error rate.[23] In addition, the high degree of heterogeneity inherent to subvisible particles and optical phenomena at small length scales further complicate the use of ConvNets to analyze FIM images. FIM images themselves can be highly variable due to flow-induced artifacts, focusing challenges, and resolution limitations.

In this work, we demonstrate a new strategy for using ConvNets for classification tasks. A simple "data pooling" strategy is combined with deep ConvNets to obtain a classifier obtaining nearly perfect performance in distinguishing various protein solutions subjected to different stresses or processing conditions using as few as 20 images to predict a class. We revisit some of the more challenging data sets explored in the study by Maddux et al.[5] to quantitatively and qualitatively demonstrate various advantages of our new ConvNet-based approach. Specifically, we reanalyze FIM images of aggregates of a mAb that were created as a result of freeze-thawing and mechanical agitation, 2 stresses that are relevant in therapeutic protein manufacturing, transportation, and drug administration.[5] In addition, we explore the use of deep ConvNet analysis of FIM images to detect differences in subvisible particles produced during simulated fill-finish operations using a fixed protein therapeutic formulation but 2 slightly different operating conditions. To do this, we created protein aggregates by pumping a single solution of intravenous immunoglobulin (IVIG) antibodies through 1 of 2 nominally identical reciprocating piston-type fill-finish pumps. We show that FIM image information alone can be used to accurately (with 0 observed error when "pooling" as few as 20 images) determine both the type of material (i.e., distinguish mAb from IVIG) and the pump used to generate aggregates within the protein solution. We additionally assessed the performance of ConvNet classifiers when relevant class labels are not present in the data used to fit the model. To perform this analysis, we trained a ConvNet to differentiate between aggregates of recombinant human interleukin-1 receptor antagonist (rhIL-1ra) and silicone oil microdroplets using FIM data taken from samples containing mixtures of the 2 components where the composition of each particle is unknown. The relevance of our new deep ConvNet approach combined with "data pooling" to protein therapeutics process monitoring and quality control is discussed. We also briefly discuss how output of this FIM image analysis approach can be combined with information from "orthogonal" measurement techniques.[3,22] In the Materials and Methods, we provide a high-level description of our ConvNet approach; detailed algorithmic details are deferred to the Supporting Information.

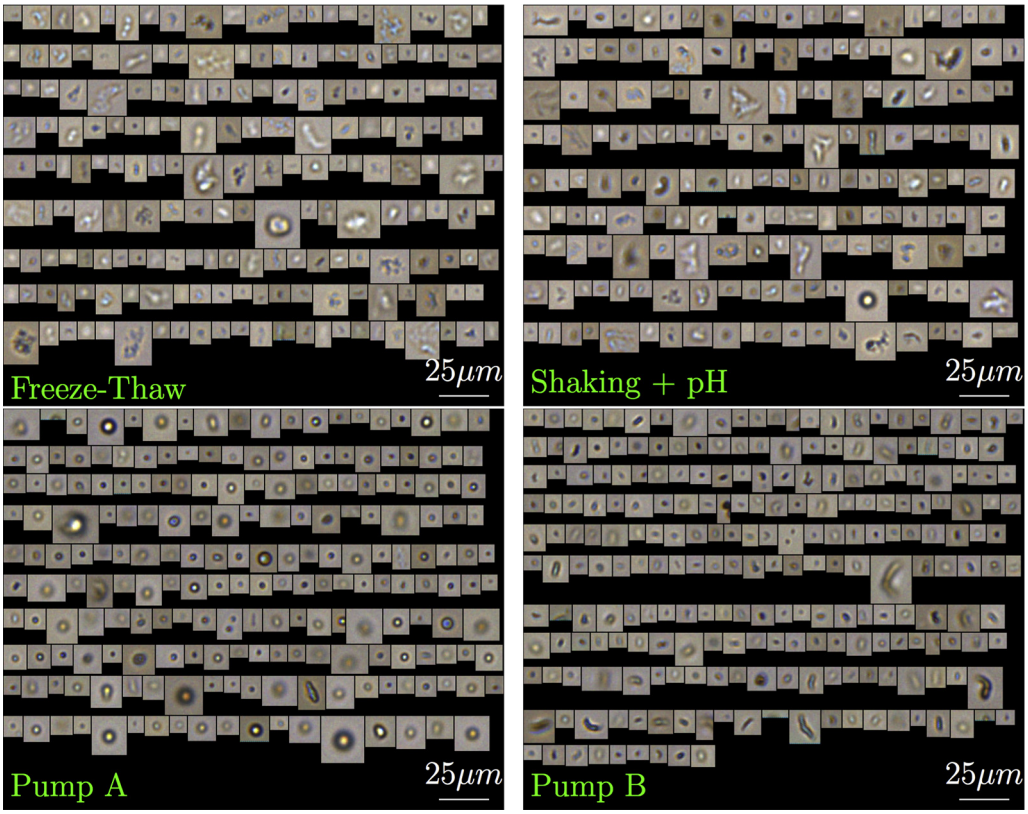

Figure 1. Sample FIM image collages from 4 FIM protein data sets. Clock-wise from top left: freeze-thawed mAb images; mAb experiencing mechanical agitation (shaking) plus pH shock; IVIG processed with a "pump A"; and IVIG processed with "pump B" (see Materials and Methods for additional sample preparation details). A ConvNet classifier was used to distinguish these 4 different conditions with high accuracy (quantitative results shown in Fig. 3). Note the heterogeneity and polydispersity of these data sets (expert humans encounter difficulty in classifying the data based on visual inspection of single images).

Materials and Methods

Materials

Lyophilized mAb was donated by Medimmune, Inc. (Gaithersburg, MD). IVIG (GAMMAGARD LIQUID) was obtained from Baxter International (Deerfield, IL). rhIL-1ra was donated by Amgen (Thousand Oaks, CA). 1 × phosphate buffered saline containing 144 mg/mL potassium phosphate monobasic, 795 mg/mL potassium phosphate dibasic, and 9000 mg/mL sodium chloride was obtained from Gibco (Waltham, MA). Dow Corning 360 medical fluid 1000 CST (Midland, MI) was used to generate silicone oil microdroplets. Hellmanex III was obtained from Hellma Analytics (Mullheim, Germany). All salts and materials used in buffer preparation were of reagent grade or higher.

Sample Preparation

A lyophilized mAb was reconstituted with water and dialyzed into 230-mM KCl at pH 6.0. The resulting mAb solution was filtered with a 0.1-μm filter and diluted to 1 mg/mL with additional KCl solution. These solutions were then exposed to freeze-thaw and shaking aggregation-inducing stresses (described below).

Freeze-Thaw Stress

Three 2-mL microcentrifuge tubes were filled with 1-mL aliquots of the mAb solution. Samples were then exposed to 10 freeze-thaw cycles. Each cycle consisted of placing the microcentrifuge tubes in a −80°C freezer for 20 min and then thawing the tubes in a water bath at room temperature for 20 min.

Agitation Stress

Three milliliters of 1 mg/mL mAb solution were dialyzed into a 20-mM citrate, 230-mM KCl solution at pH 3.0 and then immediately dialyzed again into a 230 mM KCl solution at pH 6.0. Three 1-mL aliquots were then placed into 2-mL microcentrifuge tubes and shaken horizontally at 400 rpm overnight.

Pump "A" and "B" Recirculation

IVIG of 0.5 mg/mL in phosphate buffered saline was centrifuged at 20,000 × *g* and 20°C for 20 min to remove aggregates. Forty-five milliliters of this sample was stored in 50 mL centrifuge tubes (Falcon, Corning, NY) until use. These samples were then recirculated through 1 of 2 nominally identical Filamatic FUS-10 pumps (Filamatic, Inc., Baltimore, MD), one denoted "pump A" and one denoted "pump B". The pumps were set to operate at 25 strokes per min, corresponding to a flow rate of 200 mL/min. Then, 2 mL of the sample were removed for particle analysis by FIM every minute. The samples included in this analysis were taken after 9 min of recirculation.

Protein and Silicone Oil Mixtures

rhIL-1ra was dialyzed into 100-mM phosphate buffer at pH 7.0 and diluted to 1 mg/mL using additional buffer. Three 2-mL microcentrifuge tubes were filled with 1 mL aliquots of the rhIL-1ra solution and subjected to the previously-described freeze-thaw procedure. The subvisible particle concentration of these suspensions was measured using FlowCam®. Silicone oil emulsions were generated by shearing a 8% (v/v) silicone oil in deionized water mixture at 30,000 rpm for 15 min and passing the mixture through an Emulsiflex C5 high pressure homogenizer (Avestin;

Ottawa, ON, Canada) operating at 20 MPa 7 times.[35] A The subvisible particle concentration of this emulsion was also measured using FlowCam®. Using the particle concentrations obtained for both the pure protein solution and silicone oil emulsion, aliquots of the suspensions of protein aggregates, and the suspensions of silicone oil droplets were mixed together with buffer in 2 mL microcentrifuge tubes to make 1 mL suspensions that contained approximately 50,000 particles/mL. The ratios of protein aggregates to silicone oil droplets were varied to result in a series of suspensions containing different number concentrations of protein aggregates and silicone oil droplets: 25% protein aggregates and 75% silicone oil droplets, 50% protein aggregates and 50% silicone oil droplets, and 75% protein aggregates and 25% silicone oil droplets. Three samples of each mixture were generated. Immediately after the suspensions were prepared, three 300 μL aliquots from each sample were analyzed by FIM.

Flow-Imaging Microscopy

FIM was performed with a FlowCam® VS (Fluid Imaging Technologies, Inc., Scarborough, ME) instrument equipped with a 100-mm flow cell and a 10× objective. Before use, the flow cell was cleaned with 1% Hellmanex III solution and ultrapure water. The instrument was focused using the default autofocus procedure on 20-μm calibration beads. Two hundred fifty microliters of sample mixed with 200 μL of ultrapure water were measured for each of the freeze-thaw and shaking + pH samples. Three hundred fifty microliters were measured per pump recirculation sample. The flow cell was flushed with ultrapure water between measurements.

ConvNet Structure

We trained convolutional neural networks to perform supervised classification tasks such as identifying the aggregation-inducing stress (mAb exposed to freeze-thaw, mAb exposed to agitation, IVIG circulated through pump A, or IVIG circulated through pump B) that created a given particle. Here, we provide a quick overview of the structure of these neural networks and process by which the network classifies image data. For a more in-depth review, refer study by LeCun et al.,[23] or for a textbook length treatment relevant to recent developments in ConvNets, we refer the reader to see the study by Goodfellow et al.[33]

A ConvNet sequentially pass an input image through several convolutional layers,[23] processing units designed to represent two-dimensional (2D) input such as images as a combination of several small, simple 2D patterns referred to as “filters” in the ConvNet literature. These filters are pixel patterns (estimated in a data-driven fashion) that correspond to simple features of the input. For instance, the first convolutional layer in a deep ConvNet will likely have filters that correspond to edges (e.g., the boundaries of an object like a subvisible particle) and color blobs (e.g., regions of an image that are a single color). Each of these filters in the first convolutional layer will be used to transform the raw image into a feature image, images that quantitatively encode how well different regions of the raw images match the pattern of a specific filter. For instance, the feature image generated by a filter designed to detect horizontal edges will highlight the regions of the image where horizontal object edges are present. The output of this first layer is a collection of these feature images generated from the raw image with one feature image per each filter in the convolutional layer. This collection of feature images can then be passed to the deeper layers of the network.

Modern deep convolutional neural networks architectures contain many convolutional layers (interweaved with other layer types[23]) as shown by the sample CNN in Figure 2. Each of the 3 convolutional layers in this sample network uses a unique set of filters to identify 2D patterns in the input to that layer (either the raw image or a collection of feature images obtained from the previous layer) and outputs feature images that show how well specific regions of the input match the pattern embedded in a filter. Since each convolutional layer operates on the feature images obtained in the previous layer, filters in deeper layers of the network identify increasingly sophisticated morphologic features in the initial raw image that are often difficult for a human observer to interpret. As shown in Figure 2, these convolutional layers transform a single large input image into many small feature images in the final convolutional layer of the network.[23,27] In supervised classification tasks, the feature images obtained from the final convolutional layer of the network are transformed into a large, flat vector and subsequently passed to a traditional fully connected (FC) neural network to predict the class to which the input image belongs.[33,36] It is stressed that in supervised ConvNet training, the parameters of each layer of the model are empirically learned via training, an iterative process through which images labeled with the class they belong to are used to optimize a selected cost function such as categorical cross entropy.[36] Through training, the CNN learns the filters to use in each layer to identify relevant morphologic features of the input images, which would be difficult to identify manually. These “learned” (or data-driven) morphologic features can then be used to differentiate between different image classes. By expressing the input image in terms of combinations of these filters, the CNN is able to construct a representation (i.e., a numerical expression of the features of the input) of the input image in the convolutional layers of the network that can be used to perform effective classification of the input image in the FC layer of the network. After learning the parameters of the network via training, the network can then be used to classify new images that were not used to train the network (these data are often referred to as “test data”).

ConvNet Computations

FIM images were cropped to be centered around the particle of interest and reduced to 20 × 20 RGB images in the preprocessing step (images were resized and rescaled to maintain aspect ratio). The ConvNet contained a cascade of 3 modules consisting of 2D convolutional, max pooling,[27] and dropout layers[28] connected to a FC layer. The first module consists of a convolutional layer and a max pooling layer that transforms the raw 20 × 20 RGB image to a collection of 32 9 × 9 feature images. The second module contains convolutional, max pooling, and dropout layers that transform the collection of 32 9 × 9 feature images into a collection of 32 3 × 3 features. The final module consists of a convolutional layer and a dropout layer that transforms the 32 3 × 3 feature images into 64 1 × 1 feature “images”. The FC layer changed depending on the task at hand. However, the core ConvNet module used for all computations was the same and contained a total of 28,640 trainable parameters. The full ConvNet architecture and additional implementation details are provided in Supporting Information. All ConvNet layers were implemented in Keras 2.0.6[37] (using a TensorFlow 1.2.1 backend[38]). Computations reported were carried out in an Ubuntu 16.04 Docker container environment on a machine that used 2 Nvidia GeForce GTX 1080s.

Classifying FIM Images Using ConvNets

We report results on 2 different classification problems in this section; classifying protein aggregates by the stress that was used to induce their formation, and identifying compositions of mixtures of protein aggregates and silicone oil microdroplets from mixtures of data. Both classification problems used the same ConvNet

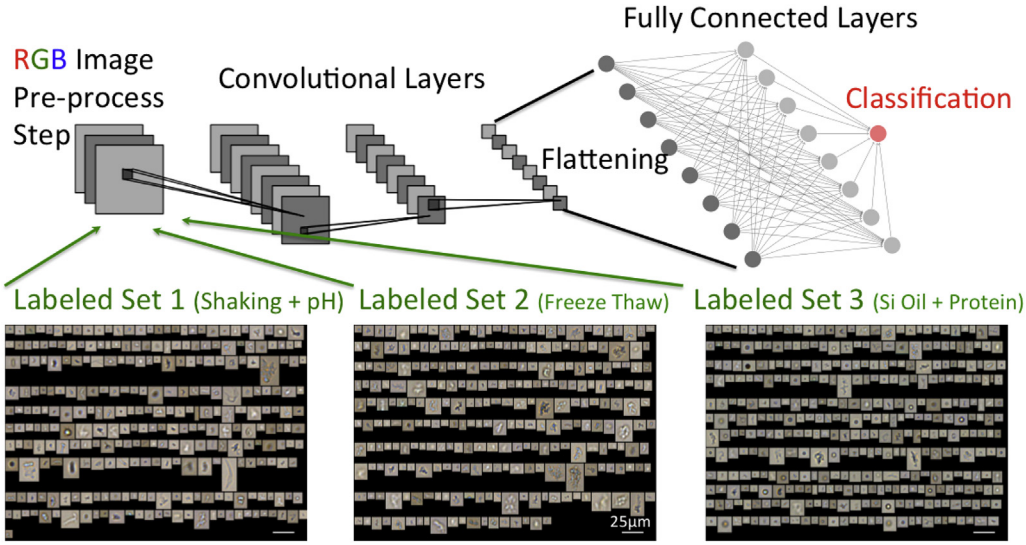

Figure 2. Illustration of ConvNet workflow. The first step preprocesses and normalizes the image (cropping, resizing, scaling, and so forth) and then passes the information to a ConvNet module. Using a large collection of "training images" processed in this fashion, the ConvNet module empirically determines a representation (or "features"[23]), which can accurately distinguish the labeled examples. After the network parameters are determined (or the ConvNet is "trained"), the network can be used to predict the labels of new samples. The FC layer above is a classic neural network.[36] In this work, we used a common ConvNet module and switched the FC layer for specific classification tasks (leveraging transfer-learning and parameter-fine tuning[25,46,47]). Figure generated with Matplotlib[48] and TikZ.[49]

module but differed in the details of their FC layers (see Supporting Information).

In both classification tasks, classification was performed both on individual images and "pools" of N_{pool} images. In pooled classification tasks, the classification prediction from a single image of our ConvNet was combined with ConvNet predictions of $N_{pool} - 1$ other single images. The median value of the pooled predictions (where predictions correspond to the output of the FC layer) served as the refined prediction result for a "block" estimate. We report results on N_{pool} ranging from 1 (single image predictions) to 100 in this work. The methodology for predicting the error in $N_{pool} > 1$ given raw test and training ConvNet data is outlined in the Supporting Information.

Classifying Aggregates by Aggregation-Inducing Stress

In the first classification problem considered, we trained our ConvNet to identify whether particles in FIM images had been generated by exposing mAb to freeze-thaw stress, by exposing mAb to agitation stress, by recirculating IVIG through filling pump A, or by recirculating IVIG through pump B. To train the model, 5×10^4 images were taken from each condition and labeled with integers 0-3 (inclusive) to indicate the experimental conditions used to generate each particle (0 indicating particles produced from IVIG in pump A, 1 indicating particles produced from IVIG in pump B, 2 indicating particles generated by agitating mAb samples, and 3 indicating particles generated by freeze-thawing mAb samples). The resulting 2×10^5 total training samples were then used to train a ConvNet model where the FC layer returned integers 0-3, which indicate the predicted class of the image (i.e., the model's prediction as to which experimental conditions were used to generate the particle). We evaluated the predictive ability of the model using 10^4 test images from each class. These images were not used to train the ConvNet, and the class label (0-3) was not presented to the model during model evaluation. We assessed the accuracy of the ConvNet classifier both on single images ($N_{pool} = 1$) as well as pools of images ($N_{pool} = 20$, 50, or 100) randomly selected from sets of images of particles all generated using the same aggregation-inducing stress. The pH and shaking data images came from multiple vials, whereas the training data for the other cases came from a single vial of solution. In the results, we present our ConvNet classifier with test images taken from new vials to demonstrate robustness of the approach.

The accuracy of the classifier using each value of N_{pool} was reported as a confusion matrix showing the computed probability that a set of test images taken from a data set associated with a given stress condition (the actual conditions are indicated by the rows of the matrix) was identified as each of the possible classes that the classifier could return (possible classes are indicated by the columns of the confusion matrix). If the classifier perfectly identified the actual class associated with all the sets of images, the diagonal values of the matrix (i.e., entries corresponding to cases where the predicted class is the same as the actual class) would be 1, and all other values would be 0.

As an additional assessment of the classifier, we studied how different values of N_{pool} affected the worst performing result from the single image ConvNet classifier (i.e., the stress that resulted in particles that were the most frequently misidentified by the classifier when $N_{pool} = 1$). To assess the improvement in accuracy caused by increasing N_{pool}, a finer grid of N_{pool} values was considered, and the classification error of the worst performing stress condition was empirically determined for the test data. These results were compared with the accuracy predicted by the uncertainty quantification approach outlined in the Supporting Information.

Characterizing Compositions of Protein Aggregate and Silicone Oil Microdroplet Mixtures

As discussed previously, the lack of "finely" labeled FIM image data sets (e.g., at the particle type level) can limit the ability to train ConvNets or other supervised machine learning techniques to perform a given classification task. Previous efforts to perform single-image classification on FIM images predominantly have been restricted to differentiating protein aggregates from silicone oil droplets, a task motivated not only by the pharmaceutical

relevance of the classification[39-41] but also by the ease of generating data sets for classifier training with FIM images of only protein aggregates or silicone oil microdroplets. However, for many other relevant particle classification tasks, it is difficult to obtain data sets where each image is labeled as a member of a particle class that we would wish to differentiate from other particle classes. In this article, we address the following question: Can we perform classification tasks such as differentiating protein aggregates and silicone oil microdroplets without access to training data sets where each image is labeled "at the particle level"?

We trained our ConvNet to differentiate between populations of particles from 2 mixtures of aggregated protein and silicone oil microdroplets. One mixture population contained 75% (by number) protein aggregates and 25% silicone oil microdroplets ("mixture 1"), and the other mixture population contained 25% protein aggregates and 75% silicone oil microdroplets ("mixture 2"), as described in the Materials and Methods section. These images were labeled according to the mixture from which they were taken (i.e., mixture 1 or mixture 2) but were not labeled as to whether each individual FIM image within the data sets was of a protein aggregate or a silicone oil microdroplet. A standard "logistical regression" approach[42] was used in this binary classification problem involving mixtures. In logistical regression, the probability that a given image belongs to a class was returned by the FC layer of the classifier instead of an integer indicating the predicted class of the input image. In this case, the classifier returns the probability that the particle in a given FIM image was taken from mixture 1 (predominantly protein aggregates). The binary category prediction can readily be determined by inspecting the probability returned by the classifier; if the predicted probability by our ConvNet was greater than 0.5, the image was assigned "class 1" (mixture 1, predominantly protein aggregates) and was otherwise assigned "class 2" (mixture 2, predominantly silicone oil microdroplets). By training the model to predict if a given image was taken from a mixed data set which predominantly contained protein aggregates, we initially hoped that the network would learn a representation of the images that could also be used to differentiate between the protein aggregates predominantly contained in mixture 1 and the silicone oil microdroplets predominantly contained in mixture 2. Although we did not expect this classifier to accurately classify single images due to the mixtures of protein and silicone oil in the training data sets, we anticipated that aggregating the classification results over several pooled images ($N_{pool} > 1$) would result in a reasonable prediction of the percentage of protein aggregates in a given pool of images.

5×10^4 images of each labeled mixture class were used to train the network to predict the mixture from which the particle was taken. We again used 1×10^4 labeled test images from each mixture that were not used in model training to evaluate the accuracy of our ConvNet classifier at identifying the mixture pools of $N_{pool} = 100$ images were drawn from as well as estimating the composition (i.e., the percent protein aggregate) of each mixture. To further test the ability of the classifier to differentiate between protein aggregate and silicone oil microdroplet, we applied our trained ConvNet to data coming from a new 50% protein and 50% silicone oil ("mixture 3") class that was not shown to the model during training. We hypothesized that due to the intermediate composition of mixture 3, the model would return an approximate 50% probability of belonging to mixture 1, corresponding to 50% protein aggregate content in the FIM images.

Results and Discussion

A ConvNet was trained to differentiate between FIM images of particles generated via 4 stress conditions. Sample FIM images of particles generated by these 4 conditions are shown in Figure 1. The performance of this ConvNet classifier for various values of N_{pool} are shown by the confusion matrices in Figure 3. The cells are "heat colored" classification probabilities (with yellow/beige indicating 0 and dark green indicating 1). A perfect classifier would have dark green along with diagonal and yellow/beige everywhere else. Although the classifier has limited accuracy when classifying an isolated image of a single particle, aggregating the predictions from several images generated under the same conditions (i.e., setting $N_{pool} = 20$) resulted in better than 95% correct classification for each class. Increasing to $N_{pool} = 100$ resulted in perfect classification in a problem where humans would have a hard time providing the correct class labels to test images like the ones shown in Figure 1. By using a simple data pooling strategy with a trained ConvNet, we were able to obtain essentially perfect classification of FIM images using a relatively small number of particle images. These results demonstrate that ConvNets can be used to successfully differentiate FIM images of aggregates on the basis of their protein composition (mAb or polyclonal IVIG) and the stress conditions used to generate the particles (freeze-thaw or agitation for mAb, 1 of 2 nominally identical pumps for IVIG). Furthermore, these results indicate that both the identity of the protein that may comprise an aggregate and the type of stress used to create the aggregate leave "morphologic fingerprints" to which FIM is sensitive and that can be identified using the ConvNet approach.

Classification tasks such as identifying the specific pump that generated a given subvisible particle are industrially relevant. For instance, many manufacturers use banks of "identical" pumps to perform fill-finish operations. If particles are found in products after fill-finish operations, it would be of general interest in a root-cause analysis investigation to determine whether the aggregates arose due to wear or malfunction in a specific fill-finish pump. Remarkably, we found that aggregates of the same protein created by passing the protein through 1 of 2 nominally identical filling pumps have morphologic fingerprints in FIM images that are sufficiently different that we can use the ConvNet analysis of only approximately 100 images to identify with essentially 100% certainty which pump created the aggregates. Thus, in this hypothetical example, it should be possible to determine exactly which pump generated the aggregates by tracking their morphologic fingerprints.

ConvNets were able to differentiate between all 4 classes of particles with over 95% classification accuracy using 20 pooled sample images and perfect accuracy using 100 pooled sample images. Although a variety of techniques have been proposed to perform similar analysis of FIM data,[5] the performance of ConvNets for this analysis is notable for 2 primary reasons. Firstly, the ConvNet classifier extracted the relevant features needed to separate these 4 classes in a purely data-driven fashion. Despite the fact that the particles were diverse and exhibited a variety of different features (see Fig. 1), the ConvNet was able to extract a representation of each image that could be used to accurately distinguish the particles. This approach uses all the information contained in the RGB images to perform classification as opposed to use only the information contained in a list of morphologic features for each particle.[3,5] Second, the deep ConvNet architecture was trained to make reasonable predictions of particle identity using only a single image. Although critical decisions would never be made on a single image in typical process monitoring and regulatory applications, the ability to obtain reasonable results on single images allows for highly accurate classifications of particle origin by pooling the results of a small number of images. Current state-of-the-art methods for distinguishing stressed mAb states require upward of 2000 "pooled images" in combination with subject-matter expert selected features to construct a highly accurate mAb aggregate

$N_{pool} = 1$

	Pump A	Pump B	Shaking + pH	Freeze-thaw
Pump A	0.70	0.30	0.01	0.00
Pump B	0.18	0.81	0.01	0.00
Shaking + pH	0.00	0.00	0.71	0.28
Freeze-thaw	0.00	0.01	0.08	0.91

$N_{pool} = 20$

	Pump A	Pump B	Shaking + pH	Freeze-thaw
Pump A	0.95	0.05	0.00	0.00
Pump B	0.01	0.99	0.00	0.00
Shaking + pH	0.00	0.00	0.98	0.02
Freeze-thaw	0.00	0.00	0.00	1.00

$N_{pool} = 50$

	Pump A	Pump B	Shaking + pH	Freeze-thaw
Pump A	0.99	0.01	0.00	0.00
Pump B	0.01	0.99	0.00	0.00
Shaking + pH	0.00	0.00	0.99	0.01
Freeze-thaw	0.00	0.00	0.00	1.00

$N_{pool} = 100$

	Pump A	Pump B	Shaking + pH	Freeze-thaw
Pump A	1.00	0.00	0.00	0.00
Pump B	0.00	1.00	0.00	0.00
Shaking + pH	0.00	0.00	1.00	0.00
Freeze-thaw	0.00	0.00	0.00	1.00

Figure 3. Confusion matrices of our ConvNet classifier tested on 10K test images from each class ("test images" are not presented to tune the ConvNet in any way) for various values of N_{pool}. The truth labels are indicated by the vertical column; the ConvNet predictions for each test image (known labels not presented to classifier) are denoted by the horizontal row (fraction of test sample with given label reported). The cells are classification probabilities (with yellow/beige indicating 0 and dark green being 1). A perfect classifier would have dark green along with diagonal. Even with $N_{pool} = 1$, the classifier performs reasonably given the heterogeneity and polydispersity of the data. With N_{pool} as small as 20, over 95% classification accuracy can be obtained for the diverse image populations tested (sample images can be observed in Fig. 1). When $N_{pool} = 100$, perfect classification for all 4 classes was obtained. Note that the same ConvNet classifier predictions reported for $N_{pool} = 1$ were "reused" with the data pooling approach discussed in the Materials and Methods.

classifier.[5] The ConvNet approach shown here only required 20 images to obtain nearly perfect mAb classification. Particle concentrations are therefore a minor concern with this approach as most FIM data sets obtained in standard protein manufacturing applications will have over 20 FIM images.

As shown in Figure 3, particles generated by pump A were the most difficult particles for the classifier to identify when $N_{pool} = 1$. Figure 4 shows both the empirically observed accuracy for differently-sized pools of images of particles generated under pump A conditions (labeled "observed") and the accuracy predicted by the uncertainty quantification approach outlined in the Supporting Information (labeled "predicted"). Note the excellent agreement in how the classifier performs when observed data is compared to the prediction. This process illustrates how a simple uncertainty quantification approach could be used to extrapolate results from a "single image" (i.e., $N_{pool} = 1$) ConvNet classification study to predict how different "pooled" results would be expected to perform—a problem of practical relevance when considering sample-size selection (in more complex problems, other more advanced statistical techniques for quantifying uncertainty could be considered[42,43]).

We emphasize that in all results reported, all "test" images were of particles not used in training the ConvNet. However, in FIM applications, protein properties can vary from vial-to-vial due to various factors including, systematic differences in protocol or the batch process generating the sample, time dependence in protein sample stability, and so forth. In our next 2 figures, we demonstrate test results from different vials to demonstrate that the CNN classifier can be extrapolated to classify protein images from different vials. Specifically, we demonstrate the accuracy of the 4-way classifier shown previously applied to test data coming from a new set of vials.

In Figure 5, we demonstrate that the results intended to mimic 2 different plants generating data, the CNN classifier is robust to both processing time and vial-to-vial variability (the vials shown were drawn from the circulation loop every minute). Empirically, we observed that the freeze-thaw protocol exhibited the highest vial-to-vial variance; these results are quantified in Figure 6. Interestingly, the replicates (i.e., analyzing new protein images from the same vial) exhibit small changes in accuracy, but vial-to-vial variance is nonnegligible. Fortunately, using N_{pool} greater than 20 (see details Supporting Information) would still produce classification results with nearly zero error. The shaking and pH studies presented earlier contained proteins from multiple vials.

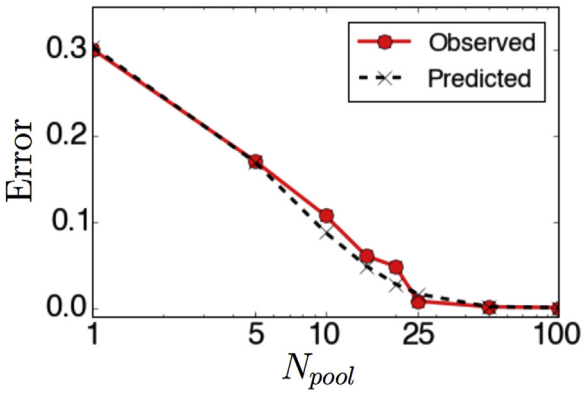

Figure 4. Predicted and observed misclassification error[43] as a function of N_{pool}. In this plot, we explored a finer grained set of N_{pool} values relative to those explored in Figure 3. The goal was to predict the misclassification error of "Pump A" as a function of N_{pool} using simple central limit theorem approximations (see Supporting Information). This type of information can be used to inform either quality control or process-monitoring applications what sample sizes are required to achieve a target false alarm or correct identification rate.

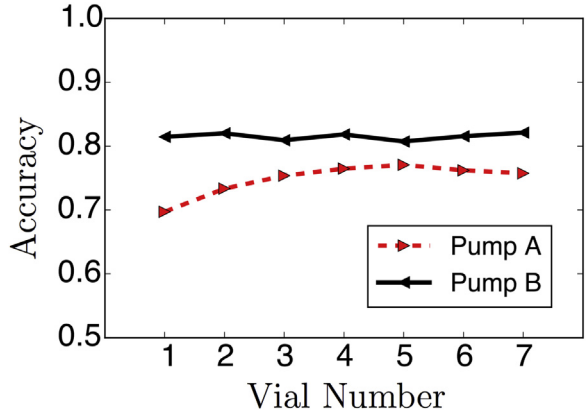

Figure 5. Results from different vials of pump A and pump B test data (i.e., images not used in training). Each marker symbol corresponds to a fresh vial and the percent of correctly labeled images in a 10K test image sample using $N_{pool} = 1$ (increasing N_{pool} improves accuracy). The vials correspond to taking different vials from the pump A and pump B data at different times. Separate images from "Vial 1" were used for training our CNN classifier (note: it is possible to draw "train" and "test" images from a single vial since each vial yielded multiple protein images). This plot demonstrates the CNN classifier is robust to vial-to-vial variability in the pump A and pump B data.

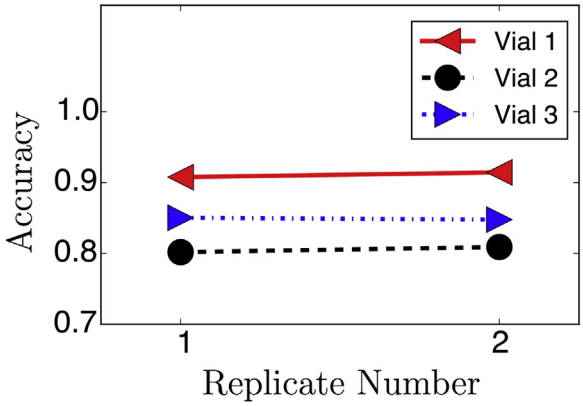

Figure 6. Analysis of different vials and replicates of freeze-thaw protein data. Each marker symbol corresponds to a fresh vial and the percent of correctly labeled images in a 10K test image sample using $N_{pool} = 1$ (increasing N_{pool} improves accuracy). Separate images from "vial 1, replicate 1" were used for training our CNN classifier (note that it is possible to draw "train" and "test" images from a single vial since each vial yielded multiple protein images). Note that using $N_{pool} = 20$ would still result in nearly error-free classification results in these test cases.

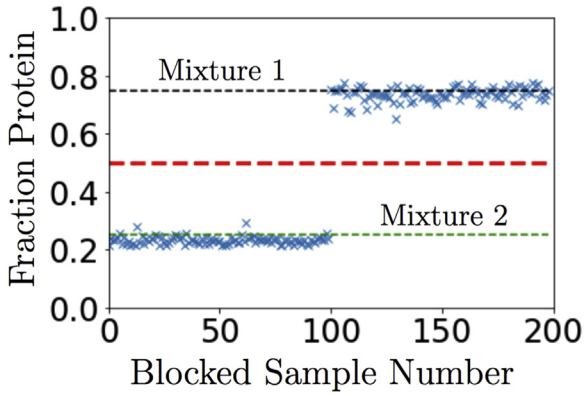

Figure 8. Error-free classification of 2 different formulations of silicone oil and protein mixtures. One formulation (mixture 1) was designed to contain 75% protein and 25% silicone oil and the other contained 25% protein and 75% silicone oil (mixture 2). 20K test samples (images not in the training set) were predicted by our ConvNet; the resulting predictions pooled into blocks of 100 resulting in 100 blocks of each sample type. The first 100 blocks of test data were mixture 2 and the last 100 blocks were mixture 1. The deep ConvNet (trained on labeled single images from these 2 mixture distributions) not only achieved perfect classification between these mixture classes with this approach but the fraction of protein (denoted by labeled horizontal lines) was almost perfectly estimated using our new deep ConvNet approach.

A separate ConvNet was trained to differentiate between 2 different mixtures of protein aggregates and silicone oil microdroplets. Sample images from these 2 mixtures are shown in Figure 7. The resulting classifier was used to estimate the probability that pools of 100 images from the 10^4 test images were taken from mixture 1. Figure 8 shows the probability returned by the network for image pools taken from both mixture 1 and mixture 2. As we suspected, the pooled output of the ConvNet classifier not only classified the 2 mixture classes perfectly but also was able to accurately predict the fraction of protein in the mixture. The results suggested that the trained ConvNet learned representations of "pure" silicone oil microdroplets and protein aggregates despite not being given these explicit labels. To confirm that the ConvNet had learned representations of both components in the mixture, we applied the ConvNet to pools of 100 images taken from mixture 3 containing an equal mixture of protein aggregates and silicone oil microdroplets. This mixture 3 should be considered a "new class" not within the training set, but if the ConvNet is approximating "pure" silicone oil and aggregates, we expect that our predictions should fall between those of mixture 1 and 2. Figure 9 shows the probabilities returned by the model when analyzing image pools taken from mixture 3. The predicted fraction of protein fell right in-between the 2 extremes with an average value of protein fraction near 50%—consistent with our hypothesis that the network had learned to differentiate between the 2 pure components. Sharper convergence to the 50% was not observed due in part to the fact that silicone oil and protein mixtures do not mix in an ideal additive fashion. Although our approach assumes that all mixtures only contain preformed protein aggregates and silicone oil droplets, proteins adsorb to the surface of silicone oil droplets, which promotes further protein aggregation. Mixing these 2 components therefore influences the particle morphology in each mixture, potentially introducing errors both in the representations learned by the network during training and the classifications given by the network when analyzing particles from mixture 3. Nonetheless, our simple ConvNet predictor performed adequately when applied to this new "mixture 3" class not represented in the training set.

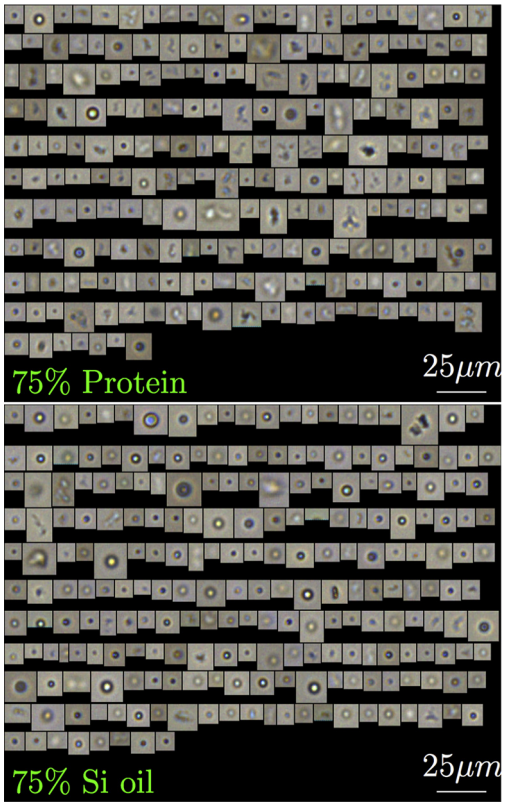

Figure 7. Sample FIM image collages from 2 of the protein/silicone oil mixture data sets.

We emphasize that our approach to this popular problem in FIM analysis[3,8] was performed on images that were not labeled as either "protein aggregate" or "silicone oil microdroplet". Instead, we trained our ConvNet on 2 mixture classes (1 mixture class was predominantly silicone oil with 25% protein "contaminant" and the other was mainly protein with 25% silicone oil "contaminant"). Although individual particle images from the labeled class could be images of a silicone oil microdroplet, a protein aggregate or some hybrid particle containing each component, our classifier was only tasked with picking the correct mixture label for each image. We demonstrated that not only could pools of 100 images be correctly classified as deriving from 1 of the 2 mixtures but also that the

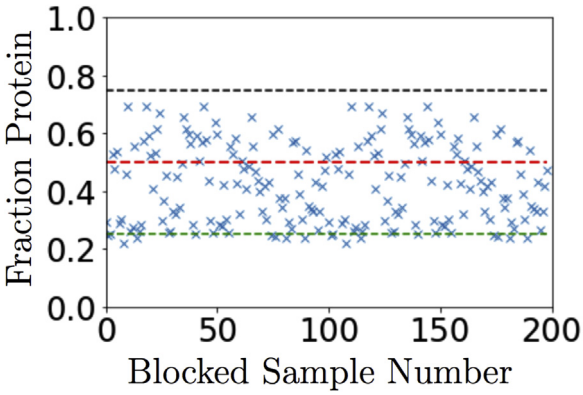

Figure 9. Two-class network trained with data shown in from Figure 5 applied to predict 10K test samples from a 50% protein and 50% silicone oil mixture. Note that the "50% protein/50% silicone oil" ("mixture 3") class was not in the training set. Extrapolating the 2-class predictor to this new mixture case consistently falls in-between the 2 training data extreme suggesting that the ConvNet has learned approximate representations of "pure protein" and "pure silicone oil" features despite these 2 cases not being explicitly labeled in individual images (the training phase only provided generic labels to the mixture type the image came from).

relative concentrations of protein aggregates and silicone oil microdroplets in both of the 2 mixtures used in training as well as in a 50/50 mixture of protein aggregates and silicone oil microdroplets could be estimated using this classifier. These results suggest that our ConvNet was able to construct approximate representations of both pure silicone oil and pure protein aggregate even in the absence of these labels in the training set. Our formulation of the protein-silicone oil classification problem is more representative of other subvisible particle characterization problems than previous formulations of this problem for the following reason: due to the inherent heterogeneity and polydispersity in subvisible particle populations it is difficult to obtain data sets where each particle is labeled by classes of interest such as particles with high immunogenic risk. These results suggest that ConvNets may be able to construct representations of protein aggregates representing risks of causing patient harm without the need to finely label every single FIM image in a given formulation.

The technology presented has potential for use in both real-time process monitoring as well as off-line analysis. Even without optimizations for speed, the Python 2.7-based code run on a single Nvidia GeForce GTX 1080 graphical processor unit could process 100 single particle images and predict the class in under 0.01 seconds in the applications presented. Advances in hardware combined with algorithms optimized for ConvNet prediction speed could reduce that time even further (hence near real-time process monitoring is feasible). We would like to note that generic neural networks are often criticized as being too "black-box" in nature (i.e., the representations these approaches use for classification are not readily human interpretable). However, ConvNets trained on image data can sometimes be human interpretable using novel visualization techniques[44]; this is an open research area in CNN research and worth intensively exploring in future FIM applications leveraging ConvNets but is beyond the scope of this work.

Finally, it should be mentioned that the ConvNets derived for a given classification problem fueled by FIM data can readily be combined with additional morphologic information about the particles such as the standard morphologic properties returned by FIM instruments. If additional morphologic parameters such as particle diameter, aspect ratio, and circularity are known to be relevant to a given classification task, they can be combined with the evaluated ConvNet features before being fed into the FC layer of the model. Combining learned features of the image with morphologic properties obtained from the instrument allow for refined classification of the particle in the final layer of the ConvNet using expert knowledge of the relevant features of the particles. In addition, ConvNet features evaluated from FIM images along with other features can be passed to another classifier capable of processing multiple types of "features" in supervised learning applications, for example, random forests.[3,45]

Acknowledgments

The authors thank Hao Wu for collecting the data on IVIG aggregates generated in the fill-finish pumps. The remaining wet-lab FIM experiments conducted by A.L.D. and designed by A.L.D. and T.W.R. were supported by NIH RO1 EB006006. The algorithm design, software development, and analysis reported here was conducted by C.P.C. and was supported by internal R&D funds of Ursa Analytics, Inc.

References

1. Carpenter JF, Randolph TW, Jiskoot W, et al. Overlooking subvisible particles in therapeutic protein products: gaps that may compromise product quality. *J Pharm Sci*. 2009;98(4):1201-1205.
2. Rosenberg AS. Effects of protein aggregates: an immunologic perspective. *AAPS J*. 2006;8(3):E501-E507.
3. Saggu M, Patel AR, Koulis T. A random forest approach for counting silicone oil droplets and protein particles in antibody formulations using flow microscopy. *Pharm Res*. 2017;34(2):479-491.
4. Kotarek J, Stuart C, De Paoli SH, et al. Subvisible particle content, formulation, and dose of an erythropoietin peptide mimetic product are associated with severe adverse postmarketing events. *J Pharm Sci*. 2016;105(3):1023-1027.
5. Maddux NR, Daniels AL, Randolph TW. Microflow imaging analyses reflect mechanisms of aggregate formation: comparing protein particle data sets using the Kullback-Leibler divergence. *J Pharm Sci*. 2017;106(5):1239-1248.
6. Joubert MK, Luo Q, Nashed-Samuel Y, Wypych J, Narhi LO. Classification and characterization of therapeutic antibody aggregates. *J Biol Chem*. 2011;286(28):25118-25133.
7. Singh SK, Afonina N, Awwad M, et al. An industry perspective on the monitoring of subvisible particles as a quality attribute for protein therapeutics. *J Pharm Sci*. 2010;99(8):3302-3321.
8. Zölls S, Weinbuch D, Wiggenhorn M, et al. Flow imaging microscopy for protein particle analysis—a comparative evaluation of four different analytical instruments. *AAPS J*. 2013;15(4):1200-1211.
9. Felsovalyi F, Janvier S, Jouffary S, Soukiassian H, Mangiagalli P. Silicone-oil-based subvisible particles: their detection, interactions, and regulation in prefilled container closure systems for biopharmaceuticals. *J Pharm Sci*. 2012;101(12):4569-4583.
10. Narhi LO, Jiang Y, Cao S, Benedek K, Shnek D. A critical review of analytical methods for subvisible and visible particles. *Curr Pharm Biotechnol*. 2009;10(4):373-381.
11. Narhi LO, Schmit J, Bechtold-Peters K, Sharma D. Classification of protein aggregates. *J Pharm Sci*. 2012;101(2):493-498.
12. *United States Pharmacopeia and National Formulary (USP29- NF24)*. Rockville, MD: USP-NF; 2007.
13. Joubert MK, Hokom M, Eakin C, et al. Highly aggregated antibody therapeutics can enhance the in vitro innate and late-stage T-cell immune responses. *J Biol Chem*. 2012;287(30):25266-25279.
14. Fradkin AH, Carpenter JF, Randolph TW. Immunogenicity of aggregates of recombinant human growth hormone in mouse models. *J Pharm Sci*. 2009;98(9):3247-3264.
15. Bessa J, Boeckle S, Beck H, et al. The immunogenicity of antibody aggregates in a novel transgenic mouse model. *Pharm Res*. 2015;32(7):2344-2359.
16. Hermeling S, Crommelin DJA, Schellekens H, Jiskoot W. Structure-immunogenicity relationships of therapeutic proteins. *Pharm Res*. 2004;21(6):897-903.
17. Jiskoot W, Kijanka G, Randolph TW, et al. Mouse models for assessing protein immunogenicity: lessons and challenges. *J Pharm Sci*. 2016;105(5):1567-1575.
18. Cavicchi RE, Carrier MJ, Cohen JB, et al. Particle shape effects on subvisible particle sizing measurements. *J Pharm Sci*. 2015;104(3):971-987.
19. Ripple DC, Hu Z. Correcting the relative bias of light obscuration and flow imaging particle counters. *Pharm Res*. 2016;33(3):653-672.
20. Cavicchi RE, Collett C, Telikepalli S, Hu Z, Carrier M, Ripple DC. Variable threshold method for determining the boundaries of imaged subvisible particles. *J Pharm Sci*. 2017;106(6):1499-1507.
21. Corvari V, Narhi LO, Spitznagel TM, et al. Subvisible (2-100 μm) particle analysis during biotherapeutic drug product development: Part 2, experience with the application of subvisible particle analysis. *Biologicals*. 2015;43(6):457-473.
22. Ríos Quiroz A, Lamerz J, Da Cunha T, et al. Factors governing the precision of subvisible particle measurement methods—a case study with a low-

concentration therapeutic protein product in a prefilled syringe. *Pharm Res*. 2016;33(2):450-461.
23. LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*. 2015;521(7553):436-444.
24. Bojarski M, Del Testa D, Dworakowski D, et al. End to end learning for self-driving cars. arXiv:160407316. 2016. Available at: http://arxiv.org/abs/1604.07316. Accessed August 1, 2017.
25. Esteva A, Kuprel B, Novoa RA, et al. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*. 2017;542(7639):115-118.
26. Zhu J-Y, Park T, Isola P, Efros AA. Unpaired image-to-image translation using cycle-consistent adversarial networks. arXiv:170310593. 2017. Available at: http://arxiv.org/abs/1703.10593. Accessed August 1, 2017.
27. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst*. 2012:1-9.
28. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res*. 2014;15:1929-1958.
29. Sergey I, Christian S. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. *32nd International Conference on Machine Learning*. Lille, France; 2015.
30. Goodfellow IJ, Warde-Farley D, Mirza M, Courville A, Bengio Y. Maxout networks. *Proc 30th Int Conf Mach Learn*. 2013;28:1319-1327.
31. Jaderberg M, Simonyan K, Zisserman A, Kavukcuoglu K. Spatial transformer networks. *NIPS Proc*. 2015;33:1242-1249.
32. Mallat S. Understanding deep convolutional networks. *Philos Trans A Math Phys Eng Sci*. 2016;374(2065):17.
33. Goodfellow I, Bengio Y, Courville A. *Deep Learning*. Cambridge, MA: MIT Press; 2016.
34. Wasserman L. *All of Statistics: A Concise Course in Statistical Inference*. Berlin, Germany: Springer Science & Business Media; 2004.
35. Gerhardt A, Bonam K, Bee JS, Carpenter JF, Randolph TW. Ionic Strength Affects Tertiary Structure and Aggregation Propensity of a monoclonal antibody adsorbed to silicone oil—water interfaces. *J Pharma*. 2012;2:429-440.
36. Bishop CM. *Neural Networks for Pattern Recognition*. New York, NY: Oxford University Press, Inc.; 1995.
37. Keras Team. Keras. Available at: https://github.com/fchollet/keras. Accessed August 1, 2017.
38. TensorFlow: large-scale machine learning on heterogeneous systems. [open source software library] 2015. Available at: tensorflow.org. Accessed July 15, 2017.
39. Chisholm CF, Nguyen BH, Soucie KR, Torres RM, Carpenter JF, Randolph TW. In vivo analysis of the potency of silicone oil microdroplets as immunological adjuvants in protein formulations. *J Pharm Sci*. 2015;104(11):3681-3690.
40. Chisholm CF, Baker AE, Soucie KR, Torres RM, Carpenter JF, Randolph TW. Silicone oil microdroplets can induce antibody responses against recombinant murine growth hormone in mice. *J Pharm Sci*. 2016;105(5):1623-1632.
41. Chisholm CF, Soucie KR, Song JS, et al. Immunogenicity of structurally perturbed hen egg lysozyme adsorbed to silicone oil microdroplets in wild-type and transgenic mouse models. *J Pharm Sci*. 2017;106(6):1519-1527.
42. Agresti A. *Categorical Data Analysis*. Hoboken, NJ: Wiley-Interscience; 2013.
43. Buonaaccorsi JP. *Measurement Error: Models, Methods, and Applications*. Boca Raton, FL: Chapman and Hall/CRC; 2010.
44. Zeiler MD, Fergus R. Visualizing and understanding convolutional networks. *Comput Vis ECCV 2014*. 2014;8689:818-833.
45. Breiman L. Random forests. *Mach Learn*. 2001;45(1):5-32.
46. Pan SJ, Yang Q. A survey on transfer learning. *IEEE Trans Knowl Data Eng*. 2010;22(10):1345-1359.
47. Yosinski J, Clune J, Bengio Y, Lipson H. How transferable are features in deep neural networks? *Adv In Neural Inf Process Syst*. 2014:3320-3328. Available at: http://arxiv.org/abs/1411.1792.
48. Hunter JD. Matplotlib: a 2D graphics environment. *Comput Sci Eng*. 2007;9(3):99-104.
49. Tantau T. *Graph drawing in TikZ, GD'12*. Berlin, Heidelberg: Springer-Verlag; 2013:517-528.