Published in final edited form as:
IEEE Int Symp Technol Homel Security HST. 2016 September 15; 2016: . doi:10.1109/THS.2016.7568959.

Adaptive Visual Sort and Summary of Micrographic Images of Nanoparticles for Forensic Analysis

Elizabeth Jurrus,
Pacific Northwest National Laboratory, Richland, WA 99352

Nathan Hodas,
Pacific Northwest National Laboratory, Richland, WA 99352

Nathan Baker,
Pacific Northwest National Laboratory, Richland, WA 99352

Tim Marrinan, and
Department of Mathematics, Colorado State University, Fort Collins, CO 80523

Mark D. Hoover
Respiratory Health Division, National Institute for Occupational Safety and Health, Morgantown, WV 26505

Abstract

Image classification of nanoparticles from scanning electron microscopes for nuclear forensic analysis is a long, time consuming process. Months of analyst time may initially be required to sift through images in order to categorize morphological characteristics associated with nanoparticle identification. Subsequent assessment of newly acquired images against identified characteristics can be equally time consuming. We present INStINCt, our Intelligent Signature Canvas, as a framework for quickly organizing image data in a web-based canvas framework that partitions images based on features derived from convolutional neural networks. This work is demonstrated using particle images from an aerosol study conducted by Pacific Northwest National Laboratory under the auspices of the U.S. Army Public Health Command to determine depleted uranium aerosol doses and risks.

I. Introduction

Forensic analysis of nanoparticles is often conducted through the collection and identification of electron microscopy images to determine the origin of suspected nuclear material. Each image is carefully studied by experts for classification of materials based on texture, shape, and size. Manually inspecting large image datasets takes enormous amounts of time. However, automatic classification of large image datasets is a challenging problem due to the complexity involved in choosing image features, the lack of training data available for effective machine learning methods, and the availability of user interfaces to parse through images. Therefore, a significant need exists for automated and semi-automated methods to help analysts perform accurate image classification in large image datasets. We present INStINCt, our Intelligent Signature Canvas, as a method for quickly organizing image data in a web-based canvas framework. Images are partitioned using small sets of example images, chosen by users, and presented in an optimal layout based on features derived from convolutional neural networks.

We demonstrate the nuclear forensic applicability of this work using particle images from the Capstone Depleted Uranium (DU) Aerosol Characterization and Risk Assessment Program [1], which a team lead by Pacific Northwest National Laboratory conducted under the auspices of the U.S. Army Public Health Command. This dataset consists of a large set of backscattered electron (BSE) images from scanning electron microscopy (SEM) of aerosol samples collected during perforation of an Abrams tank and a Bradley vehicle with DU munitions. The details of aerosol collection in the high-energy environment of the Capstone study have been described by Holmes et al. [2]. The motivation for examining the images of the particles is to determine the particle morphology, especially in the nano-size range. This morphology provides insight into the relationship between the chemical formation, the solubility, and the dissolution rates [1] present during their formation. As reported by Krupka et al. [3], ultrafine aerosols of aluminum and iron from the vehicle armor were observed in the analysis of the Capstone particle images. The mechanism of formation for such particles in the nano-size range from high temperature processes involves vaporization-condensation mechanisms. Guilmette and Parkhurst further reported that little to no evidence was obtained that demonstrated the existence of DU nanoparticles [4]. Examples of BSE images of a selection of the particles from the Capstone study can be seen in Figure 1.

II. Methods

Our INStINCt framework was developed to enable efficient organization of SEM images through the integration of a web-based interface, state-of-the-art image feature vectors, and a new clustering technique. The user interacts with the INStINCt interface through a light weight web client application that provides a virtual canvas for organizing images and enabling different users to collaborate on the same datasets. The canvas displays images represented by a 4096-dimensional feature vector computed from OverFeat, a convolutional neural network[5]. Since we cannot easily visualize these high dimensional feature vectors, we attempt to map these feature vectors into a 2-dimensional space while preserving their class relations. The method we implemented to perform dimensionality reduction is an adaptive spatialization method. First, the user-selected images are clustered using DBSCAN [6] and “anchors” for each group are computed. The positions of the remaining images are updated in the 2-dimensional spatialization using a coordinates determined by the similarity of the images to these clusters. Similarity is measured by the angle between a feature vector and the low-dimensional subspace created from the anchor. The relationships between unlabeled images and small, user-selected clusters of images are visualized via diffusion map in the canvas. One of the advantages of our method for representing the clusters is that in the inevitable case that the visualization does not represent all of the semantic information recognized by the domain expert, the visualization can be globally updated by moving a small number of samples closer to or further from the clusters.

A. Client Interface

Central to our implementation is our web-based software for displaying images and recording human operator interactions. We leverage our activeCanvas web application to display the computed visualizations. The activeCanvas allows image files to be uploaded to a server, from which they can be organized into a 2D layout by means of JSON files that contain most relevantly, file names, (x, y)-coordinates, and a Boolean flag that indicates whether or not an image was moved on the canvas. The activeCanvas includes the ability to zoom and pan across the canvas as well as an image magnifier that allows users to see an enlarged version of the currently selected image. It also includes the ability to select multiple images as a group to quickly arrange images on screen as the user desires according to the task at hand.

Via a REST-ful interface, the user sends the JSON file, containing new (x, y)-coordinates and updated “moved” Boolean flags for each image, to the server. The server analyzes only the images the user moved, and, according to the methods outlined in Section II-C, computes a new location for all images on the screen. The updated (x, y)-coordinates are placed into the JSON, and that information is pushed back to the web-interface. Images on the screen smoothly translate to the new positions calculated by the server. The full translation takes about 1.5 seconds, providing the user with a visual trace of the changes. The user then adjusts additional images according to their tasks, based on the new positions, to provide additional input to the prediction algorithm. The cycle repeats until the user is satisfied with all the positions of the images on the screen, and then the user may export the JSON to another application for further analysis, such as building a classifier. Additionally, users can upload supplementary images as part of the main workflow, adding additional data to the layout if needed.

B. Feature Selection

Feature selection for computer vision is a challenging problem. There are many methods for finding relevant features in images and creating feature vectors to represent each image. Some of the most relevant work in this area focuses on keypoint detection and aggregation of similar keypoints to identify similarities across images [7]. Instead of trying to find the best keypoint detector for our data, we chose to represent each image with a 4096-dimensional feature vector that is computed as the output of Sermanet *et al.* ’s convolutional neural network called OverFeat [5]. OverFeat was exhaustively trained to identify object boundaries on a massive corpus of real-life images from the ImageNet Large Scale Visual Recognition Challenge [8]. This training allows it to effectively extract feature vectors that would allow nearly linear separation between the object classes in the 4096-dimensional space. It turns out it also acts as a very good general feature descriptor for image data [9]. We found that when we performed a qualitative evaluation of the OverFeat features against standard image descriptors, the OverFeat features provided better separation of different particle images.

C. Dimensionality Reduction

As a result of the mapping from the original pixel space of the images to the feature space via the convolutional neural network, the SEM data should be close to linearly separable in

$\mathbb{R}^{4096}$. However visualization in 4096 dimensions is not feasible, so we attempt to preserve class relations in a reduced dimension visualization. Most low dimensional mappings look to preserve pairwise relationships between all data points either locally or globally under the assumption that little is known about the data itself. Conversely, our application domain is centered around a human operator who has working knowledge of the data, and some relevant relationships. The operator's goal is to explore other hypothetical relationships in the data or to quickly organize the images with respect to the known, relevant information rather than to all possible information.

With this task in mind, we look to display points based on their relationship to user selected clusters of data. Starting from a naive 2-dimensional layout of the image data, a user specifies a few small collections of related images by moving them into separate, spatially contained regions of the visualization. The initial layout presented to the user could be as simple as a tessellation of the images or randomly generated (x, y)-coordinates. We would like to illustrate the improvements that our adaptive method for visual sort and summary provides over existing techniques, so we generate our initial layout using the well-known nonlinear embedding diffusion map technique [10].

The diffusion map layout is created by constructing a Gaussian kernel from the distances between data points; a technique that works equally well in Euclidean space or on a manifold equipped with a metric. Distances for this kernel will be measured using the minimum principal angle between two subspaces. As described by Björck and Golub, principal angles can be computed for subspaces of different dimensions, and specific to our application it will work for measuring the angle between one feature vector and a subspace generated from the span of a collection of feature vectors [11].

This Gaussian kernel is used to compute a normalized graph Laplacian, and each row of the Laplacian is rescaled to unit length so as to find the 1-step transition probabilities. The eigenvectors corresponding to the largest magnitude eigenvalues of this matrix then represent the coordinates of the low dimensional embedding, with the exception that the first eigenvector is constant and is thus dismissed. Figure 2 shows the initial layout of the Capstone particle images using a standard 1-step diffusion map.

The dimensionality reduction performed in the INStINCt framework is cued by user-created clusters. Thus our description here will follow this process. As a simple example, suppose a user manually creates two small clusters of related images by moving them into distinct, spatially constrained areas of the activeCanvas as is shown by the circled images in Figure 3. The interaction with the activeCanvas is recorded using a JSON file that indicates the new (x, y)-coordinates of the moved images as well as the Boolean flag showing that they have been touched. This file can be exported from the activeCanvas if further analysis and computations are desired.

Because of an intentional limitation in our architecture, we are not allowed knowledge of the previous locations of the images. Thus we cannot determine the user's intentions strictly based on an image's updated coordinates. For example, we would not know that two images have been moved closer together from the information in the JSON file. We would only know that both images were moved and their current coordinates. Instead, we interpret the human operators actions by clustering the images that have been moved based on their 2-dimensional coordinates. For this task we employ the Density-based spatial clustering of applications with noise (DBSCAN) algorithm of Ester *et al.* [6]. The DBSCAN method looks for areas of high density in the data to create clusters. DBSCAN was chosen because it does not require knowledge about the number of clusters ahead of time, it can find clusters of arbitrary shape, and it allows for some points to remain unclustered if they are not located in dense regions. The final point is advantageous because sometimes the user wishes to inject class information by pushing dissimilar images further apart, rather than by grouping similar images. Thus the INStINCt framework does not require that an image's movement away from a cluster necessarily forces that image into another cluster. One additional note, the DBSCAN method assumes that clusters are of relatively equal size, so skewed group sizes generated by a user may yield less desirable results. However, since we assume that the human operator has only moved a small number of points, this should not be an issue.

The DBSCAN clustering step determines which images a user thinks are similar. The inferred knowledge from the clusters is then used to create a subspace "anchor" as a representative for each similarity group. An anchor consists of a subspace average of the feature vectors for the images in that cluster. We impose the constraint that the space spanned by each anchor must not overlap with the anchors for any of the other clusters. Additionally, moved images that were not assigned to a cluster are represented in the anchor of the nearest cluster with a reduced weight proportional to their distance from that group.

For instance, assume that a human operator moved some images into two dense clusters, and moved a few other images into a non-clustered region away from those sets. The anchors, U_A and U_B, for clusters A and B are computed as follows. Let

$$\left\{a_i\right\}_{i=1}^{N} \text{ and } \left\{b_j\right\}_{j=1}^{M} \quad (1)$$

be the collections feature vectors associated with the images in cluster A and cluster B respectively, such that a_i, b_j $\mathbb{R}^{4096}$ for all i, j. Let

$$\left\{x_k\right\}_{k \in K} \quad (2)$$

be the feature vectors from images that were not placed in a cluster by DBSCAN, and

$$\left\{x_a\right\}_{a \in K} \quad (3)$$

be the subset of those points that are closer in 2-dimensional Euclidean distance to cluster A than to cluster B. For all the vectors in $\left\{a_i\right\}_{i=1}^{N}$ and $\{x_a\}_{a \in K}$, project away from the span of vectors in cluster B. That is,

$$\widetilde{a}_i = \left(I - PP^T\right)a_i \quad (4)$$

where P is an orthonormal basis for the span of $\{b_1, b_2, \ldots, b_M\}$.

Associate a weight of $\lambda_i = 1$ with each element of $\left\{\widetilde{a}_i\right\}_{i=1}^{N}$ and

$$\lambda_a = \frac{1}{\min\left\{\left\|x_a - a_i\right\|_2 \text{for } i = 1\ldots N\right\}} \quad (5)$$

with each element of $\left\{\widetilde{x}_a\right\}_{a \in K}$. The anchor U_A is then computed as the weighted flag mean of this input data using the method described by Draper *et al.* [12]. The anchor U_B is computed for cluster B in an analogous fashion.

Using the subspace anchors that represent each of the spatial clusters, we now compute the principal angle between each feature vector in the data set and each of the anchors. Let s be the number of points in the data set and let r be the number of clusters that the human operator has formed. Suppose Y is a square, symmetric matrix of dimension $(s + r)$ $(s + r)$. Populate the elements in the lower left $r \times s$ block of Y, and symmetrically the upper right $s \times r$ block of Y, with the principal angles between the feature vectors and the anchors. The values in the upper left $s \times s$ submatrix are set to a small constant, ε, and the values lower right $r \times r$ submatrix are set to a relatively large constant α. From this symmetric matrix, we compute the 1-step diffusion map to get the updated (x, y)-coordinates of the visualization.

The result of setting the pairwise distances of the images, i.e. the upper left block of Y, to a small constant is that images want to collapse to a single point in the diffusion map embedding. Only their relative distances to the anchors pull them apart. Similarly, the large constant value set for the lower right block of Y attempts to push the distinct clusters away from each other. As we will show in the following Section (and in Figure 4), this provides something like a gradient between the different types of particles in our dataset.

III. Results

In our application domain we assume that the human operator has some knowledge about the data, and they are looking to explore other hypothetical relationships or to organize the data relative to this known information rather than to all the information available. Thus, we are only looking to display points based on their relationship to some user selected clusters of data. Figure 2 shows an initial layout of the particle image data using a standard 1-step diffusion map. From this initial layout, the user manually creates two small clusters of related images by moving them in the INStINCt framework, as is shown by the circled images in Figure 3. The purpose of the clustering step is to determine which images the user thinks are related. Once that information is gathered, we create an “anchor” to represent each cluster. The anchor consists of an average of the feature vector representations of the images in that cluster. The remaining images are arranged according to their distances to those anchors. The results of the associated update is shown in Figure 4. If we use three user created clusters, we see a layout like the one in Figure 5.

These initial results using clustering indicate there is a strong correlation between shape and texture and particle type. However, this data is missing labels and requires an approach that combines clustering on feature data with input from the user to perform semi-automated classification on particle types.

IV. Conclusions

This paper presents a new method for user analysis of particle image data produced by SEM. The work presented in this paper represents an exploration into an adaptive data visualization technique through the representation of high dimensional spaces. Our results appear to reflect semantic relationships in a way that is hopefully valuable to human operators. Users choose to indicate relationships, via the positions chosen on the client canvas, which are indicative of shape, size, and texture. We envision this tool as a unique processing capability for triaging large quantities of images, either to provide comparisons across benchmarked datasets or simply qualifying groups of similar particles.

There are a number of directions for extending this research in future work. First, we would like to incorporate multiple feature vectors for each particle image. For the case of the data set at hand, each particle has been imaged exactly once and therefore corresponds to a single feature vector. However, the method implemented in INStINCt is flexible enough to handle the span of multiple feature vectors as a representative for a single particle. This situation could arise if the particles were additionally analyzed as high-resolution secondary electron images. We also would like to take into account the scale of each particle to incorporate size to our analysis. Finally, we would like to extend this proof-of-concept to include a database that would enable more flexibility for saving user input, such as image groupings, morphological labels, and possible explicit relationships.

Acknowledgments

The research described in this paper is part of the Signature Discovery Initiative at Pacific Northwest National Laboratory (PNNL). It was conducted under the Laboratory Directed Research and Development Program at PNNL, a multiprogram national laboratory operated by Battelle for the U.S. Department of Energy. The authors also gratefully acknowledge the work of the U.S. Army Public Health Command and the PNNL-lead study team related to the Capstone DU Aerosol Study, with special appreciation to David Alberth for assistance with the particles images from that study which were evaluated in this paper. The findings and conclusions in this paper are those of the authors and do not necessarily represent the views of Pacific Northwest National Laboratory, Colorado State University, or the National Institute for Occupational Safety and Health.

References

1. Parkhurst M, Daxon E, Lodde G, Szrom F, Guilmette R, Roszell L, Falo G, McKee C. Depleted uranium aerosol doses and risks: Summary of U S assessments. Battelle Press; 2005.
2. Holmes TD, Guilmette RA, Cheng YS, Parkhurst MA, Hoover MD. Aerosol sampling system for collection of capstone depleted uranium particles in a high-energy environment. Health Physics. 2009; 96(3):221–237. [PubMed: 19204482]

3. Krupka KM, Parkhurst MA, Gold K, Arey BW, Jenson ED, Guilmette RA. Physicochemical characterization of capstone depleted uranium aerosols iii: Morphologic and chemical oxide analyses. Health Physics. 2009; 96(3):276–291. [PubMed: 19204486]
4. Lykken GI, Momcilovic B. Comment on the capstone depleted uranium (DU) aerosol characterization and risk assessment study. Health Physics. 2010; 98(1):77. [PubMed: 19959955]
5. Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, Le-Cun Y. Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv: 1312.6229. 2013
6. Ester M, Kriegel H-P, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. KDD. 1996; 96:226–231.
7. Szeliski R. Computer vision: Algorithms and applications. Springer Science & Business Media; 2010.
8. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, et al. Imagenet large scale visual recognition challenge. International Journal of Computer Vision. 2014:1–42.
9. Razavian A SharifAzizpour H, Sullivan J, Carlsson S. CNN features off-the-shelf: An astounding baseline for recognition. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops; June 2014;
10. Coifman RR, Lafon S. Diffusion maps. Applied and Computational Harmonic Analysis. 2006; 21(1):5–30.
11. Björck Ȧ, Golub GH. Numerical methods for computing angles between linear subspaces. Mathematics of Computation. 1973; 27(123):579–594.
12. Draper B, Kirby M, Marks J, Marrinan T, Peterson C. A flag representation for finite collections of subspaces of mixed dimensions. Linear Algebra and its Applications. 2014; 451:15–32.

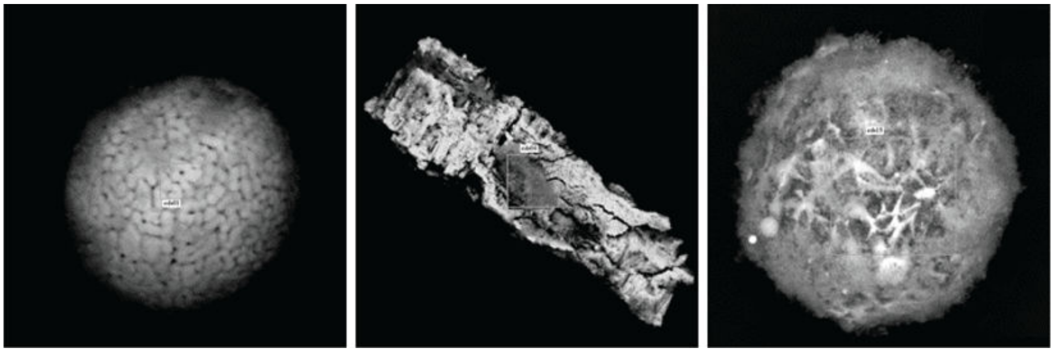

Fig. 1.
Example particle images from the Capstone Aerosol Study.

Fig. 2.
Initial diffusion map layout of the particle image data.

Fig. 3.

Particle image data with two user created clusters of images.

Fig. 4.
Particle image data with two clusters after the distance-to-anchor diffusion map update.

Fig. 5.
An example layout for the particle image data after the user selects three clusters and the distance-to-anchor diffusion map update.