Supporting Information

Seeking regularity from irregularity: Unveiling the synthesis–nanomorphology relationships of heterogeneous nanomaterials using unsupervised machine learning

Lehan Yao[a], Hyosung An[a,b], Shan Zhou[a], Ahyoung Kim[a], Erik Luijten[c,d,e,f], Qian Chen[a,g,h,i]*

[a]Department of Materials Science and Engineering, [g]Department of Chemistry, [h]Materials Research Laboratory, [i]Beckman Institute for Advanced Science and Technology, University of Illinois, Urbana, IL 61801, United States.

[b]Current address: Department of Petrochemical Materials Engineering, Chonnam National University, Yeosu, 59631, Korea.

[c]Department of Materials Science and Engineering, [d]Department of Engineering Sciences and Applied Mathematics, [e]Department of Chemistry, [f]Department of Physics and Astronomy, Northwestern University, Evanston, IL 60208, United States.

*Correspondence to: qchen20@illinois.edu

Materials and Methods

No unexpected or unusually high safety hazards were encountered.

1. Chemicals

1.1 Chemicals for gold tetrahedra nanoparticle and patchy gold nanoprism synthesis

Gold(III) chloride trihydrate (≥49.0%, $HAuCl_4 \cdot 3H_2O$, Sigma-Aldrich), cetyltrimethylammonium chloride (CTAC) (>95%, $C_{19}H_{42}ClN$, TCI), sodium iodide (99.999%, NaI, Sigma-Aldrich), L-ascorbic acid (AA) (BioXtra, ≥99.0%, Sigma-Aldrich), sodium hydroxide (99.99%, NaOH, Sigma-Aldrich), cetyltrimethylammonium bromide (CTAB) (BioXtra, ≥99.0%, $C_{19}H_{42}BrN$, Sigma-Aldrich), sodium borohydride (98 %, $NaBH_4$, Sigma-Aldrich), polystyrene-block-poly(acrylic acid) (PS-*b*-PAA) (PS_{154}-*b*-PAA_{49}, M_n=16,000 for the PS block and M_n=3,500 for the PAA block, M_w/M_n=1.15, Polymer Source Inc.), 2-naphthalenethiol (2-NAT, 99%, Sigma-Aldrich), N,N-dimethylformamide (DMF) (anhydrous, 99.8%, Sigma-Aldrich), acetone (≥99.5%, Fisher Chemical), and isopropanol (99.9%, Fisher Chemical) were purchased and used without further purification. Water used in this work was nanopure water (18.2 MΩ·cm at 25 °C) purified by a Milli-Q Advantage A10 system.

1.2 Chemicals for polyamide membrane synthesis

Cadmium chloride hydrate (99.998%, $CdCl_2 \cdot xH_2O$, $x \approx 2.5$, Alfa Aesar), ethanolamine (>98%, Sigma-Aldrich), m-phenylenediamine (MPD, 99%, Sigma-Aldrich), 1,3,5-benzenetricarbonyl trichloride (a.k.a. trimesoyl chloride, TMC, 98%, Sigma-Aldrich), molecular sieves (3 Å, 1–2 mm beads, Alfa Aesar), hydrochloric acid (36.5–38.0%, HCl, Macron), polysulfone film (PS35, Sepro Corporation), potassium hydroxide (KOH), hexanes (99.9%, Fisher Chemical) and isopropyl alcohol (IPA, Macron) were purchased and used without further purification. Approximately 225 g molecular sieves were baked at 110°C in a glass jar for 1 day and then stored with hexane (1 L). All glassware was cleaned with a base bath (saturated KOH in IPA), followed by an acid bath (1 M HCl), thoroughly rinsed with water, and dried with nitrogen gas. Cadmium chloride hydrate, MPD, and TMC were carefully stored in a desiccator to prevent exposure to moisture, which is important for reproducible membrane synthesis.

2. Material synthesis

2.1 Synthesis of gold tetrahedral nanoparticles

The gold tetrahedra used here were synthesized following a literature method.[1] Spherical seeds with a diameter of 8 nm were first synthesized based on the seed-mediated growth and collected by centrifugation at 14,500 rpm for 30 min, and then dispersed in 1 mL of aqueous CTAC solution (20 mM) for further use. Aqueous solutions of CTAC (200 mM, 0.75 mL), CTAB (100 mM, 0.5 mL), AA (100 mM, 1.0 mL), and the spherical seed solution with the seed concentration characterized by the optical density (OD) of the solution at the maximum extinction wavelength in UV-Vis spectra of 521 nm (OD = 1.39, 24.0 µL, 39.7 µL, and 44.0 µL for different reaction conditions) were mixed with water (746.0 µL, 730.3 µL, and 726.0 µL for three reaction conditions, respectively) in a 20 mL glass vial to make the final volume 0.75 mL, followed by dropwise addition of aqueous $HauCl_4$ solution (0.5 mM, 1 mL) using a syringe pump at an injection rate of 0.5 mL h^{-1}. The reaction was conducted on an orbital shaker at a speed of 400 rpm and allowed to proceed at room temperature for 10 min after the injection had been completed. The final product was collected by centrifugation at 8000 rpm for 10 min. After the centrifugation, the supernatant was removed as much as possible, and the sediment (50 µL) was re-dispersed by adding 0.9 mL water into the centrifuge tube. An aliquot of the nanoparticle suspension (0.3 µL) was drop-casted onto the transmission electron microscopy (TEM) grid and air dried before the TEM characterization.

2.2 Synthesis of patchy gold nanoprisms

The patchy gold nanoprisms were synthesized following a literature method.[2] The prisms with the maximum extinction wavelength in UV-Vis spectra (λ_{max}) of 655 nm were prepared and purified as a stock prism solution, which had an OD of 5 and ≤ 0.005 mM CTAB. For the synthesis of small lobed patchy prisms, a 2-NAT solution (5 µL, 2 mg/mL in DMF) was mixed with 815 µL DMF in an 8 mL glass vial. 100 µL of a stock prism solution and 100 µL water were sequentially added dropwise into the vial using pipette and on a vortex, followed by adding PS-*b*-PAA solution (80 µL, 8 mg/mL in DMF) in one shot without vortexing. The α of the reaction mixture, defined as the ratio of 2-NAT molar concentration to OD of the prisms at λ_{max}, is 0.12 mM. The vial was tightly capped with a Teflon-lined cap and sonicated for 5 s, parafilm-sealed, and heated at 110 °C in an oil bath and left undisturbed for 2 h. The vials were slowly cooled down to the room temperature in the oil bath, which typically took 90 min. The solution was transferred to a 1.5 mL microcentrifuge tube and centrifuged three times (at 4500 rpm, 4200 rpm, and 3600 rpm for 15 min each) to separate the residual 2-NAT and PS-*b*-PAA in solution from the as-prepared patchy nanoprisms. After each centrifugation, 1.45 mL of the supernatant was removed and the 50 µL sediment was re-dispersed with 1.45 mL of water. After the third round of centrifugation, 5 µL of the 50 µL sediment containing patchy prisms was drop-casted on a transmission electron microscopy grid pretreated with oxygen plasma (30 sec at low power), then left to dry in the air for 3 hrs. For the patchy prisms synthesized at different ligand concentrations, α was increased to 0.25, 0.5, 1.0, and 2.0 mM, respectively, at a fixed c_{pspaa} of 30 nM in the reaction mixture, as detailed in Table S1.

2.3 Synthesis of polyamide membranes

Polyamide membranes were synthesized as described previously.[3] A sacrificial support layer of cadmium hydroxide ($Cd(OH)_2$) nanowires was prepared by sequentially adding aqueous solutions of $CdCl_2 \cdot xH_2O$ (50 mL, 4 mM) and ethanolamine (50 mL, 2 mM) to an Erlenmeyer flask (250 mL). The solution was stirred with a Teflon-coated magnetic stir bar (2 cm in length) at 500 rpm at room temperature for 15 min and turned cloudy. The polysulfone substrate (6 cm × 6 cm), having been stored in water for 12 h, was fixed on a glass filter funnel (3.8 cm inner diameter) connected to a filtering flask, which was connected to a vacuum pump (KNF, UN726.3 FTP). The polysulfone substrate was washed by filtering 20 mL methanol, followed by filtering 50 mL water under vacuum. The solution of $Cd(OH)_2$ nanowires was filtered across the polysulfone substrate under vacuum (−67 kPa). An aqueous solution of MPD with the desired concentration (c_{MPD}, from 1 to 5 w/v%) was then gently transferred onto the nanowire-coated polysulfone using a 10 mL micropipette and filtered under vacuum (−67 kPa). Hereafter, we will use % to designate only weight per volume percent (gram per mL, w/v%). A TMC solution in hexane with the desired concentration (c_{TMC}, from 0.05 to 1%) was gently transferred onto the polysulfone substrate. Interfacial polymerization initiated immediately upon contact of the MPD and TMC solutions. After 60 seconds, the TMC solution was gently removed using a micropipette and then pure hexane (10 mL) was added to rinse away excess TMC. This rinsing step was repeated total three times. Immediately afterwards, the polysulfone substrate covered with the synthesized polyamide membrane was placed in a water-filled Petri dish (10 cm in diameter). The polyamide membrane separated from the polysulfone substrate and floated at the air–water interface. The water in the Petri dish was replaced with a 10 mM HCl solution using a micropipette. The polyamide membrane was kept floating on the dilute HCl solution overnight to remove any residue of the $Cd(OH)_2$ nanowires. The HCl solution was then replaced with water five times. The membrane was scooped onto a carbon film-coated TEM grid and was dried in air for TEM imaging.

3. TEM characterization methods

A JEOL 2100 Cryo TEM with a LaB_6 emitter at 200 kV was used for taking images of gold tetrahedral nanoparticles synthesis products and of patchy gold nanoprisms, and for the tomography of polyamide membranes. For tomography, low electron dose rates (4–7 e^{-} Å^{-2} s^{-1}) were applied using spot size 3 to minimize beam-induced alteration. We prepared five polyamide membranes with c_{MPD} and c_{TMC} systematically varied: c_{TMC} of 0.05%, 0.1%, or 1% with a fixed c_{MPD} of 1%; and c_{TMC} of 0.05% or 0.1% with a fixed c_{MPD} of 2%. For each polyamide film, a total of 61 tilt images were acquired over a tilt range of –60° to +60° with angle increment of 2°. Each image was collected with an exposure time of 1 s, resulting in a dose per image of 4–7 e^{-} $Å^{-2}$. The sample was set to its eucentric height at each tilt angle manually, followed by a defocus of −2048 nm to improve contrast, and the same defocus was used throughout all tilt series acquisitions. TEM images were aligned and assembled using the patch tracking module in the open-source software IMOD 4.9.3 (University of Colorado, http://bio3d.colorado.edu/).[4] After the alignment, the tomograms of these samples were generated using the Model-Based Iterative Reconstruction (MBIR) algorithm with a diffuseness of 0.3 and a smoothness of 0.2.[5]

Image Processing and Data Analysis

4. Tetrahedral nanoparticle

4.1 Image preprocessing

In total 16 experimental TEM images in dm3 format with a 2048×2048-pixel resolution are split into 144 images in tiff format with a 512×512-pixel resolution through MATLAB to match the input size of the neural network, and the intensity values are normalized by the highest and lowest value on the central column and row of the image.

4.2 Neural network Training and prediction

For training data, our previous work on liquid-phase TEM videos using U-Net relied on image simulation to produce “synthetic” data with well-defined ground truth. Due to the complication of diffraction and phase contrast in dry TEM images and the difficulty in modeling impurities, here we show an alternative method for generating a training data set based on image augmentation, which is convenient and widely applicable. 18 cropped experimental images are picked and manually labeled to generate the training dataset. We use ImageJ to manually label the overlaying pixels and non-overlaying pixels and they are saved separately in two channels in RGB images, which serve as ground truth and are used for the augmentation (Fig. S2a). The augmentation is done through the Keras library in Python, where we apply random rotation, zooming, and flipping to both experimental images and labeled ground truths and also brightness variation to the experimental images only to generate 1000 image pairs for U-Net training (Fig. S2b). The random combinations of these augmentation operations give rise to a large number of new images, which are sufficient for the U-Net training. After the augmentation, a third channel—the negation of the union of the first two channels—is added to the labels to represent the background pixels. A randomly initialized U-Net is trained with an Adam optimizer (learning rate = 10^{-4}) and a validation split of 0.2 in the Keras library on Google Colab. The training is stopped at 50 epochs, which gives training loss, training accuracy, validation loss, and validation accuracy of 0.0298, 98.69%, 0.0470, and 98.27%, respectively. Subsequently, we pass all experimental TEM images to the trained U-Net to obtain the prediction (Figs. S3a,b). The predictions are binarized by assigning 1 to each pixel in each channel if the predicted probability of this pixel in this specific channel is the highest among 3 predicted probabilities of this pixel in all 3 channels, and otherwise 0. Figure S3b shows the recolored binarized predictions.

4.3 Particle contour reconstruction

Figure S9a demonstrates the nanoparticle contour reconstruction workflow. The isolated areas in the channel of non-overlaying pixels are separated by a watershed algorithm to split the nanoparticles still connected in this channel, while the areas touching the image boundary are also removed. Subsequently, for each newly isolated area in the non-overlaying channel, we compute the union of the area and the entire overlaying channel. The union is separated by the watershed again, and the isolated area containing each original non-overlaying area is recorded as a separate nanoparticle contour.

This method works well when the overlaying region is small and simple (Fig. S3). For more complicated cases, the segmentation identifies unphysical particle contours (Fig. S3c). We have summarized the scenarios for reconstruction of the overlaying particle contours. 1. The detection method can recognize particles when the overlaying region only involves two particles. When a third or more particles stack on top of two particles, the detection fails due to the increased combinations of possible ways for all the three particles to join the overlaying and non-overlaying regions (Fig. S3, second column). 2. For one pair of particles to be separated, there has to be at least one overlaying region. In other words, when two particles are seamlessly in touch with each other without overlaying, no overlaying regions can be predicted by the U-Net. Consequently, the method recognizes the two touching yet non-overlaying particles as a single particle (Fig. S3, third column). 3. The overlaying region between two particles needs to be darker than the non-overlaying region, which is usually so due to increased Z-contrast. In the cases when the Z-contrast is overwhelmed by diffraction contrast of crystalline material in TEM imaging, wrong prediction from U-Net of particle contours could be resulted (Fig. S3, fourth column). 4. The overlaying region cannot be too large. In our observation, large (*e.g.*, > 50%) overlaying region leads to no prediction of the overlaying region although the Z-contrast is clear (Fig. S3, fifth column). We suspect the U-Net model develops such behavior to prevent from predicting random dark features such as those due to diffraction contrast within one particle as the overlaying regions.

Failed detections count for around 10% of the nanoparticles detected, which are manually removed prior to further analysis for accurate yield counting (Fig. S3d). A total of 2,753 nanoparticle contours, including both tetrahedral nanoparticles and impurities with random shapes, are extracted from the TEM images. Removing the failed detections has minor impact on the product tetrahedral particle characterization, which will be discussed in detail in Supporting Information Section 4.5.

4.4 Shape fingerprint extraction

The shape fingerprint $d(\theta)$ is computed for all contours. Vectors with angles θ from –179° to 180° are generated at an interval of 1°, emanating from the geometrical centroid of each contour. The distance $d(\theta)$ traveled by each vector within the nanoparticle contour is recorded as a function of θ. For a convex shape, this method produces the same results as the literature[6] definition of $d(\theta)$ (where the $d(\theta)$ is the centroid-to-contour distance), while for slightly concave shapes, it describes the contour as well as the concaveness. Each shape fingerprint is shifted on the θ-axis so that the highest $d(\theta)$ value corresponds to $\theta = 0$. Using a small angular interval allows us to retain more shape details, especially for the particles imaged at high magnification.

4.5 PCA and GMM

The shape fingerprints $d(\theta)$ of 2753 contour samples are dimensionally reduced by PCA; the cumulative variance of the top 50 principal components is shown in Fig. S6a. We use the two largest principal components (Fig. S6c) to serve as the input for a Gaussian Mixture Model (GMM) classifier with six centers. We found that both three centers and six centers are elbow points, suggested by the Bayesian (BIC) or Akaike information criterion (AIC) (Fig. S6b). However, when we select three centers, the model is unable to identify the three obvious clusters (cf. Fig. S6c), so that we decided to use six centers for the GMM. The GMM clustering and BIC/AIC calculation are performed using the scikit-learn library in Python and apart from the number of clusters, all input parameters are kept at their default values. The BIC and AIC values are averaged over 20 GMMs with random initializations. The clustering result before combining the impurity classes is shown in Fig. S6d. We found that clusters 1, 4, and 6 represent impurity particles, and therefore combined them into one class in Fig. 1f.

The shape fingerprints $d(\theta)$ of all 3,131 contours without manual selection are also dimensionally reduced by PCA, which gives data point distribution with three obvious clusters (Fig. S10a), similar to the PCA of contours after removing the failed detections (Fig. S6c). Then a GMM classifier with eight centers reproduces the three major classes representing the tetrahedral nanoparticle products synthesized under three different reaction conditions (Fig. S10b), which are almost identical to the classification result from the data with failed detections removed (Fig. S6d). The edge lengths and truncations are measured from the clustering results with and without removing the failed detections respectively to confirm their similarity. As the results show in Table S2, in our method, not removing the failed detections does not have obvious impact on the edge length and truncation measurements of the gold tetrahedral nanoparticle.

4.6 Edge length and truncation measurement of tetrahedral nanoparticles

The tetrahedral nanoparticle contours are truncated (or rounded) at the tip regions. To measure the edge lengths of the particles, we first remove points on the contour near each tip. Then we perform orthogonal regression on the rest of points belonging to each straight edge to fit an ideal triangle without truncation (Fig. S11a). The fitted lines show good quality and consistency across all contours (Fig. S11a). The average edge length of each fitted triangle is then defined as the edge length of the tetrahedral nanoparticle (Fig. S11b).

The truncation is expressed as the distance from the tip of the fitted ideal triangle to the interception between the tetrahedral nanoparticle contour and the line connecting the same tip of the fitted ideal triangle and the triangle centroid (Fig. S11b).

The edge length and truncation distributions of gold tetrahedral nanoparticles synthesized at different seed amount are plotted in Fig. S11c.

4.7 Additional evaluations of the prediction

Other than the reconstruction failures which are discussed in Sections 4.3 and 4.5, the accuracy of the contour reconstruction is also evaluated by comparing the shape fingerprint, edge length, and truncation produced by the contours reconstructed from the manually labeled ground truth and the model prediction (Fig. S10 c–g). By plotting the values in the shape fingerprints, edge lengths and truncations of ground truth versus those of U-Net prediction, we show that most data points concentrate on the diagonal line, which means the good consistency between the ground truth and U-Net predictions (Fig. S10 e–g). Several outliers are observed in the plots, which can be back tracked to their corresponding contours as highlighted in the panel c and d in Fig. S10. Such outlier contours are due to the failed watershed segmentation step during the reconstruction and should be removed in the workflow. Here we keep them to evaluate the impact of reconstruction failures on the shape fingerprint, edge length, and truncation quantities. The root-mean-square-errors of shape fingerprints, edge lengths, and truncations between the ground truth and prediction are calculated to be 0.3 nm, 0.4 nm, and 0.4 nm respectively, which show the sub-nanometer precision of our method.

5. Patchy nanoprisms

5.1 Image preprocessing

Experimental TEM images in dm3 format with a 2048×2048-pixel resolution are scaled and saved in tiff format with a 512×512-pixel resolution in MATLAB to match the input size of the neural network. The intensity values in each image are normalized by the intensity standard deviation of each image.

5.2 Neural network training and prediction

The training data set generation and augmentation follow the same procedure as in Section 4.2. Pixels representing gold nanoprism core and the polymer patch are labeled for 12 experimental images (Fig. S2c) and then augmented to generate 1200 image pairs for training (Fig. S2d). A U-Net with the same architecture and the same training parameters as in Section 4.2 is trained on the augmented image–label pairs for 20 epochs, yielding a training loss, training accuracy, validation loss, and validation accuracy of 0.0130, 99.48%, 0.0141, and 99.45% respectively. Subsequently, we feed all experimental TEM images to the trained U-Net, binarizing the prediction as in Section 4.2 (Figs. S4a,b).

5.3 Contour and shape fingerprint extraction

While most experimental TEM images of the patchy nanoprisms contain only one particle, some contain more than one. To count the correct number of patchy nanoprisms, for each image we first compute the union of the binarized prediction channels representing the nanoparticle core and the polymer patches. Then for each isolated area in the union (i.e., each patchy nanoprism instance), the intersection between this area and the entire nanoparticle core channel is identified as the core of this nanoprism, while the intersection between this area and the entire polymer patch channel is identified as the polymer patches of the nanoprism. Following the method above, we collect and centralize all 321 non-aggregated patchy particle shapes. Illustrative examples are shown in Fig. S4c.

To focus on the shape of the polymer patch on each tip of the nanoprism, for one patchy nanoprism, we first extract the shape fingerprint of the nanoprism core $d_{\mathrm{prism}}(\theta)$ following the same procedure as in Section 4.4 and determine the angle θ^* corresponding to the maximum $d_{\mathrm{prism}}(\theta)$. The binarized nanoprism image is rotated over $-\theta^*$ so that the tip with maximum $d_{\mathrm{prism}}(\theta)$ points in the $\theta = 0$ direction. The shape fingerprint of all polymer patches is then created using the same method as in Section 4.4, except that the $d(\theta)$ only accounts for the polymer patches, without considering the nanoprism core. The $d(\theta)$ is split into three fingerprints centered at $\theta = 0°$, $\theta = 120°$ and $\theta = -120°$ respectively, each with an angular range of 120° and describing the patch morphology of one tip (Fig. 2e). Some polymer patches that are asymmetric with respect to the prism tip can exhibit mirrored fingerprints; to resolve this ambiguity, we flip fingerprints with respect to $\theta = 0°$ such that the higher $d(\theta)$ value for an asymmetric patch is always located on the right-hand side ($\theta > 0$).

5.4 PCA and GMM

The shape fingerprints $d(\theta)$ of 963 contour samples are dimensionally reduced by PCA following the same method as in Section 4.5, with the cumulative variance of the top 50 principal components shown in Fig. S7a. We use the three largest principal components (Figs. S7c,d) as the input for a GMM classifier with six centers, as a local minimum and an elbow point suggested by BIC and AIC curves (Figs. S7b). The GMM clustering follows the same method as described in Section 4.5. The clustering result is shown in Figs. S7e,f and Fig. 2f.

5.5 Patch coverage and asymmetry measurement

We define the patch coverage as the angular range over which $d(\theta)$ is larger than 0.5 nm. The asymmetry of a patch relative to the prism tip is determined from the average absolute difference between the shape fingerprint $d(\theta)$ and its mirrored counterpart,

$$\mathrm{asymmetry} = \frac{\sum_\theta |d(\theta) - d(-\theta)|}{\sum_\theta 1} \tag{1}$$

Both patch coverage and asymmetry are first measured from every individual shape fingerprint and then averaged from the GMM predicted classes to obtain the mean value and standard deviation (Fig. S7g).

5.6 The impact of material contrast on prediction accuracy

F or broader interests in predicting materials with different contrast under TEM, the contrasts of labeled patchy nanoprism images are adjusted to test the trained U-Net model (Fig. S12). During the contrast adjustment, the intensity values above the background value and below the gold nanoprism value are fixed while the rest of intensity values are adjusted via gamma correction. The values are first mapped in the range 0–1 and then assigned by $I = I_0{}^{\gamma}$, where I and I_0 are new and original intensity values respectively and γ represents the gamma value. Gamma values < 1 make polymer intensity close to the background intensity and gamma values >1 make polymer intensity close to the gold prism intensity (Fig. S12a). The results show that the U-Net model used in this work makes good predictions (Fig. S12b) with accuracy higher than 98% (Fig. S12c) for gamma values in between 0.6 and 2. For extreme gamma values < 0.2 or > 6, the loss of polymer patch or gold prism core are observed (Fig. S12b).

Such behavior is expected because the training dataset is aimed for the experimental TEM images which does not contain gamma correction or polymer contrast variation. The robustness of U-Net against the contrast variation can be easily gained by including the contrast variation in the training dataset. Figures S12d,e show the prediction results of the retrained U-Net, which give superior accuracies above 98% for all gamma values we include in the test. In practice, including images of materials with different contrasts in the training dataset should be a plausible way of training U-Net models with good performance.

6. Polyamide membrane crumples

6.1 Segmentation of 3D polyamide membrane tomographs

Segmentation of the reconstructed membrane tomographs is performed in Amira 6.4 (FEI).[7] For segmentation and 3D morphology analysis, the reconstructed tomograms are imported into Amira. For segmentation, a median filter with 3×3×3 neighborhood voxel in 3D and 26 iterations is applied, followed by a 3D Gaussian filter with kernel size 9 in 3D and standard deviation voxel of 3×3×3 and subsequently a 3D edge-preserving smoothing filter with 25 time, 5 step, 3.5 contrast, and 3 sigma. Contrast and brightness are not adjusted. A grayscale threshold is set on a per-sample basis to generate an approximately segmented volume, which is then corrected using a semi-manual adjustment to fill in holes or remove regions not corresponding to the crumple. This segmentation procedure is applied to all five polyamide membrane samples.

6.2 Shape fingerprint extraction

The binary volumetric images of individual crumples were used for the shape fingerprint extraction. Similar as in Sections 4.4 and 5.3, vectors were generated emanating from the 3D geometrical centroid of the crumple with polar angles θ from –90° to 90° and azimuthal angles φ from 1° to 360°, both with an interval of 2°. The distances traveled by these vectors within the crumple geometry $d(\theta, \varphi)$ were recorded as a function of θ and φ (Fig. 3b). The 3D fingerprint extraction was performed by customized MATLAB codes.

6.3 PCA and GMM

The shape fingerprints $d(\theta, \varphi)$ of 140 individual crumples are dimensionally reduced by PCA. The shape fingerprints $d(\theta, \varphi)$, which are 90×180 matrices, are first rearranged into vectors with 16200 elements. The resulting vectors are dimensionally reduced by PCA in MATLAB. Due to an insufficient number of observations, only 140 principal components can be computed, with the cumulative variance of the first 50 principal components shown in Fig. S8b. We use the three largest principal components (Figs. S8d,e), which account for 73% of the total variance, as input for a GMM classifier with four centers, which is roughly based on our observation of the data points distribution. Even though both the BIC and AIC curves

(Fig. S8c) suggest three centers, we increased this to four to distinguish more crumple morphologies (empirically, we found that using three centers would simply combine the "lay" and "pancake" classes) The GMM clustering and BIC/AIC calculation were done using the scikit-learn library in Python and apart from the number of clusters, all input parameters were kept at their default value. The BIC and AIC values were averaged across 20 GMMs with random initializations. Figures 3d,e visualize the clustering results from different viewing angles and Figs. S13a–d show all 140 shape fingerprints grouped according to the classes identified by the GMM.

6.4 Extraction of surface curvature elements

To extract crumple shape descriptors, individual crumples are cropped from the segmented tomograms. Amira[7] is used for the analysis of crumple surface curvatures. Segmented crumples are converted to a network of triangular meshes using the Generate Surface function with 1/16 simplification. The meshed network is smoothed using the Smooth Surface function with four iterations and lambda = 0.7. Principal curvatures (κ_1, κ_2) are defined as $1/R_1$ and $1/R_2$, where R_1 and R_2 are the radii of the smallest and largest spheres that can be fit to the surface at each mesh point. Curvature elements (tube, saddle, tip, valley) at each mesh point are determined using an Arithmetic function,

$$(\kappa_1 > 0 \;\&\&\; \kappa_2 > i) + 2(\kappa_1 > 0 \;\&\&\; \kappa_2 < i \;\&\&\; \kappa_2 < j) + 3(\kappa_1 > 0 \;\&\&\; \kappa_2 < j) + 4(\kappa_1 < 0 \;\&\&\; \kappa_2 < j) \tag{2}$$

where && denotes the Boolean "and." This expression returns 1 for a "tip" element, 2 for a "tube" element, 3 for a "saddle" element, and 4 for a "valley" element. The thresholds i and j were chosen as 0.005 and −0.005 because the second principal curvature at "tube" elements is never perfectly zero (i.e., the film is never perfectly flat).

References

1 Y. Zheng, W. Liu, T. Lv, M. Luo, H. Hu, P. Lu, S.-I. Choi, C. Zhang, J. Tao, Y. Zhu, Z.-Y. Li and Y. Xia, Seed-mediated synthesis of gold tetrahedra in high purity and with tunable, well-controlled sizes, *Chem. Asian J.*, 2014, **9**, 2635–2640.

2 A. Kim, S. Zhou, L. Yao, S. Ni, B. Luo, C. E. Sing and Q. Chen, Tip-patched nanoprisms from formation of ligand islands, *J. Am. Chem. Soc.*, 2019, **141**, 11796–11800.

3 S. Karan, Z. Jiang and A. G. Livingston, Sub–10 nm polyamide nanofilms with ultrafast solvent transport for molecular separation, *Science*, 2015, **348**, 1347–1351.

4 J. R. Kremer, D. N. Mastronarde and J. R. McIntosh, Computer visualization of three-dimensional image data using IMOD, *J. Struct. Biol.*, 1996, **116**, 71–76.

5 Z. Wang, S. Liang, Y. Jin, L. Zhao and L. Hu, Controlling structure and properties of polyamide nanofilms by varying amines diffusivity in organic phase, *J. Membr. Sci.*, 2019, **574**, 1–9.

6 C. R. Laramy, K. A. Brown, M. N. O'Brien and Chad. A. Mirkin, High-throughput, algorithmic determination of nanoparticle structure from electron microscopy images, *ACS Nano*, 2015, **9**, 12488–12495.

7 D. Stalling, M. Westerhoff and H.-C. Hege, in *Visualization Handbook*, eds. C. D. Hansen and C. R. Johnson, Butterworth-Heinemann, Burlington, 2005, pp. 749–767.

Table S1. Reaction conditions for synthesizing patchy prisms with different patch shapes

α (mM)	volume of stock solution II (μL)	volume of water (μL)	volume of DMF (μL)	volume of 2-NAT solution (2 mg/mL in DMF, μL)	volume of PS-*b*-PAA solution (8 mg/mL in DMF, μL)
0.12	100	100	815	5	80
0.25	100	100	810	10	80
0.5	100	100	800	20	80
1.0	100	100	780	40	80
2.5	25	175	795	25	80

Table S2. Edge length and truncation measurements of gold tetrahedral nanoparticles with and without manually removing the failed detections

Class	Property	With removal	Without removal
1	Edge length (nm)	46.1 ± 0.6	46.1 ± 0.8
2		40.5 ± 0.7	40.5 ± 1.1
3		37.8 ± 0.6	37.8 ± 0.6
1	Truncation (nm)	7.0 ± 0.4	7.1 ± 0.6
2		5.6 ± 0.7	5.7 ± 0.9
3		5.9 ± 0.4	6.0 ± 0.4

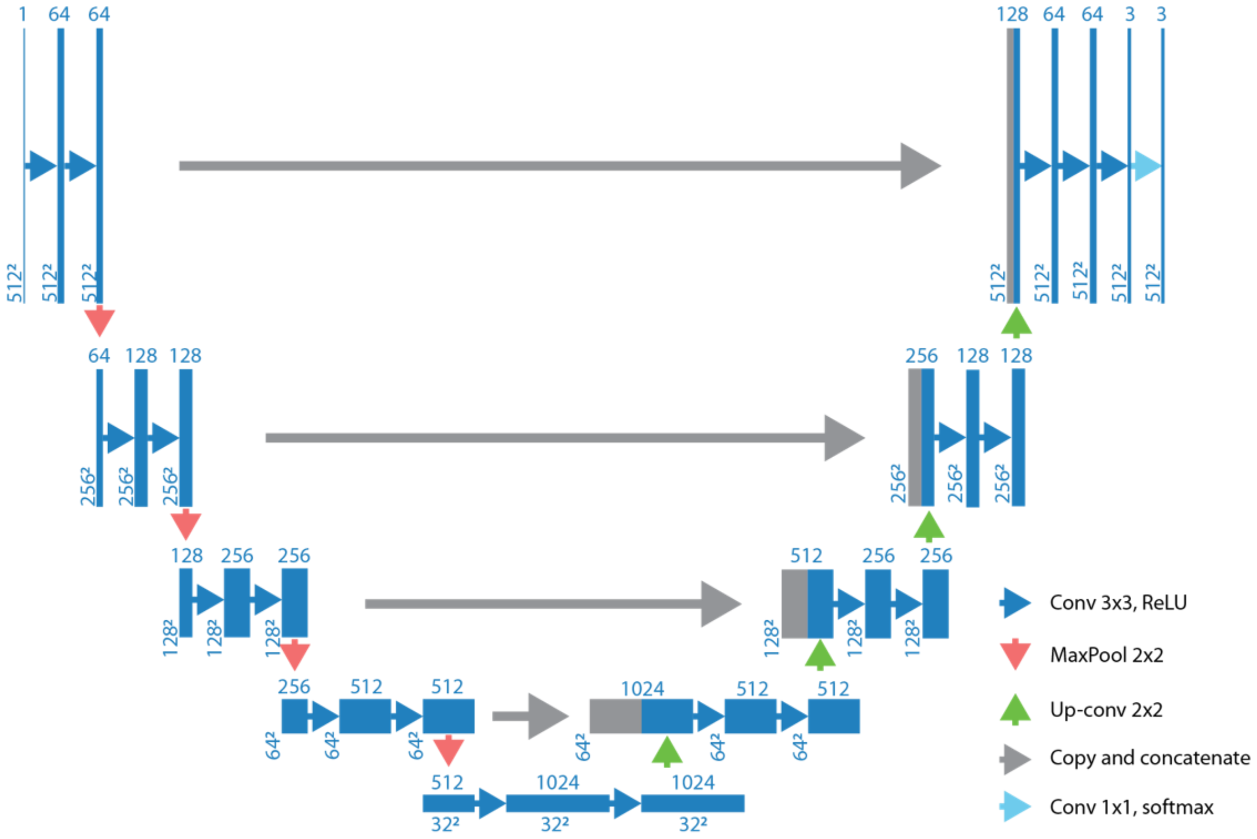

Figure S1. Schematic of the U-Net architecture with three output channels used in this work.

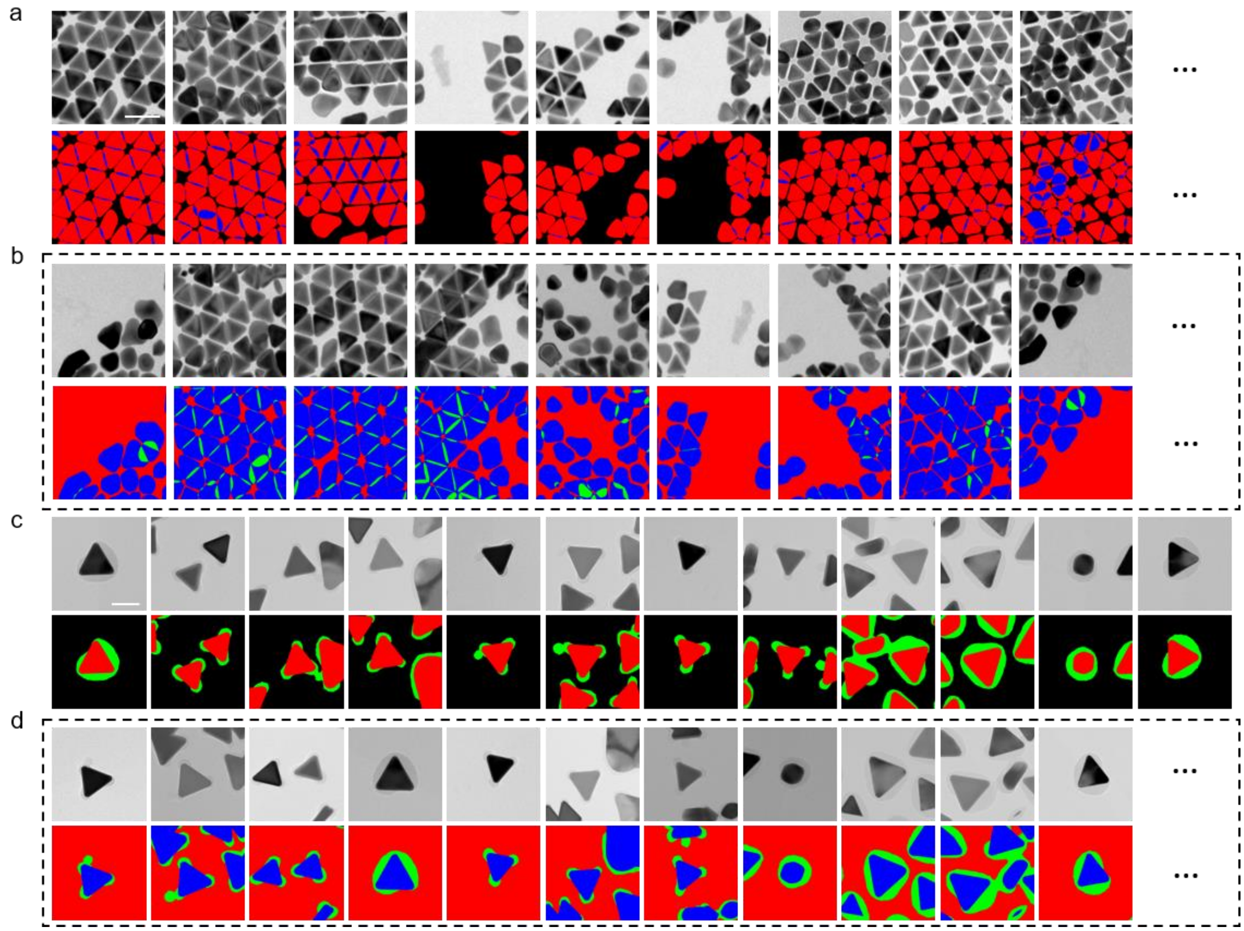

Figure S2. (a) Examples of the manually labeled experimental TEM images of the gold tetrahedra synthesis products. Top panel: TEM images. Bottom panel: labeled images. Red represents non-overlaying region; blue represents overlaying region. (b) Examples of the augmented training data set of the TEM images of the gold tetrahedra synthesis products. Top panel: augmented TEM images. Bottom panel: augmented labels. Red represents background; blue represents non-overlaying region; green represents overlaying region. (c) Examples of the manually labeled experimental TEM images of patchy gold nanoprisms. Top panel: TEM images. Bottom panel: labeled images. Red and green represent gold nanoprisms and polymer patches, respectively. (d) Examples of the augmented training data set of the TEM images of the patchy gold nanoprisms. Top panel: augmented TEM images. Bottom panel: augmented labels. Red, blue, green represent background, gold nanoprism, and polymer patch, respectively. Scale bars: 50 nm.

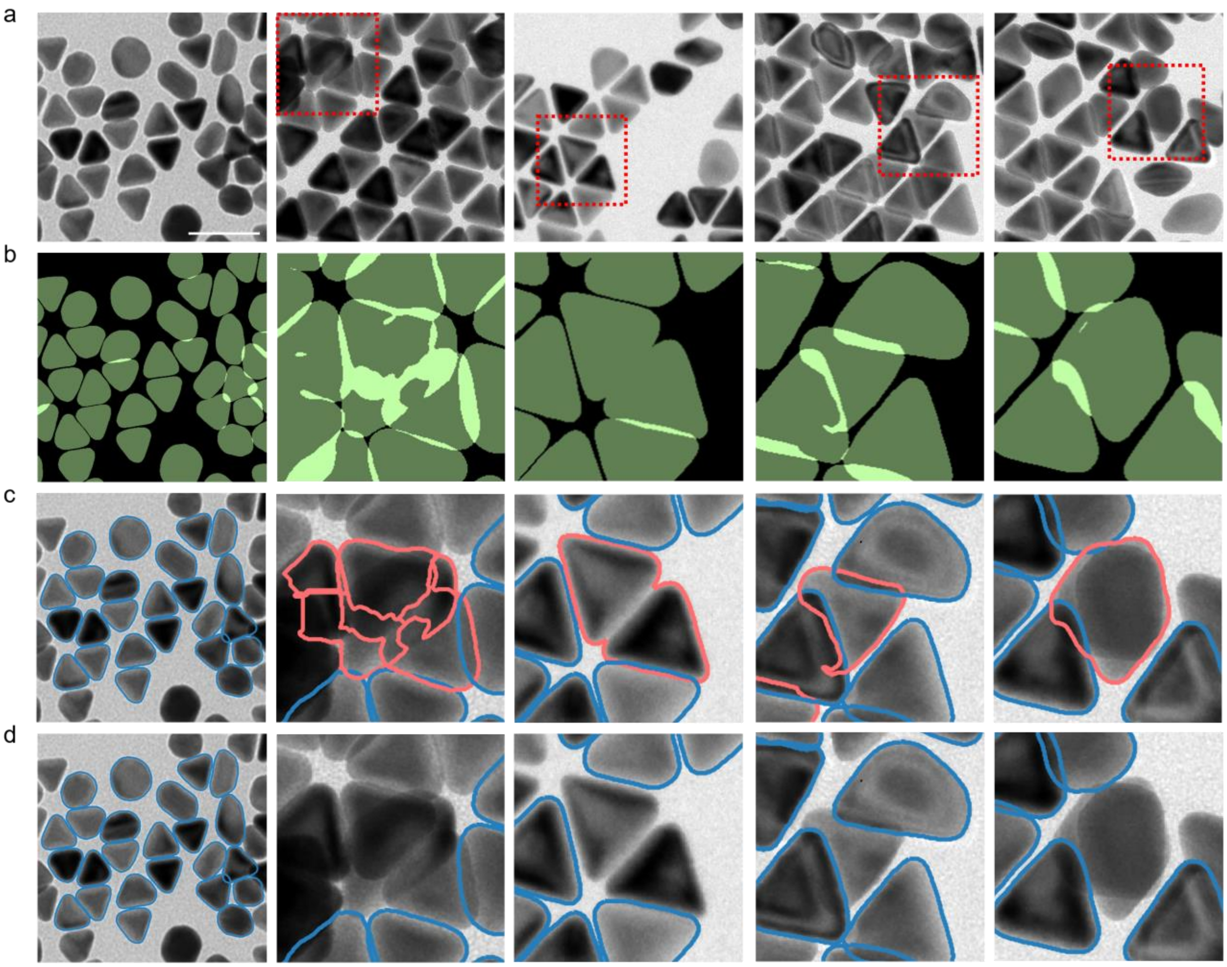

Figure S3. (a) Examples of experimental TEM images of the gold tetrahedra synthesis products. (b) U-Net predictions corresponding to (a) or the cropped regions of (a) delineated by the red dashed line box. Dark green represents non-overlaying region; light green represents overlaying region. (c) Nanoparticle contours reconstructed from (b) by our algorithm. Blue color annotates good reconstructions. Red color annotates failed detections. (d) Resulting nanoparticle contours after the failed detections have been removed manually. Scale bar: 50 nm.

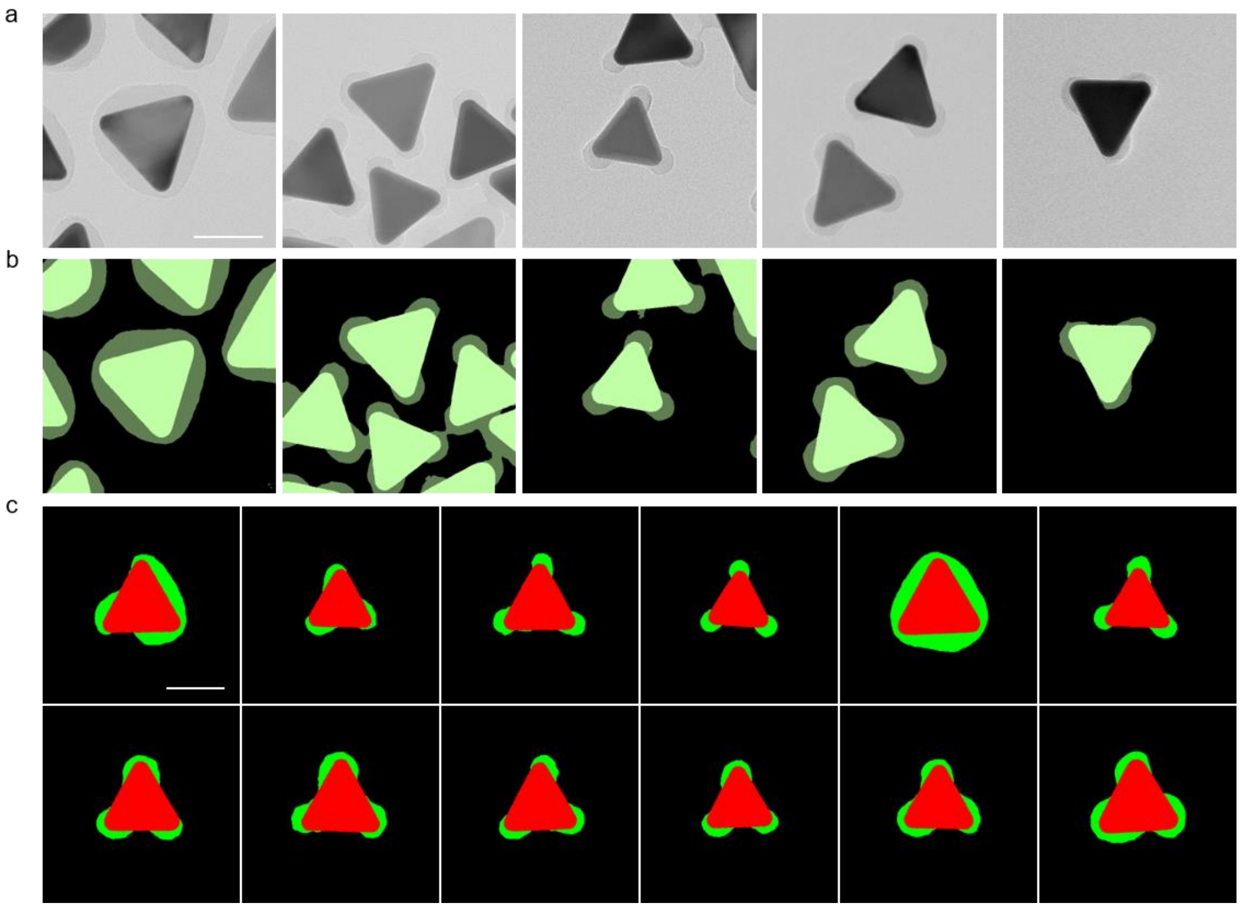

Figure S4. (a) Examples of experimental TEM images of the patchy gold nanoprisms. (b) U-Net predictions corresponding to (a). Polymer patch (dark green); gold nanoprism (light green). (c) Examples of centered and reorientated individual patchy nanoprism shapes extracted from the experimental TEM images. Green represents polymer patch; red represents gold nanoprism. Scale bars: 50 nm.

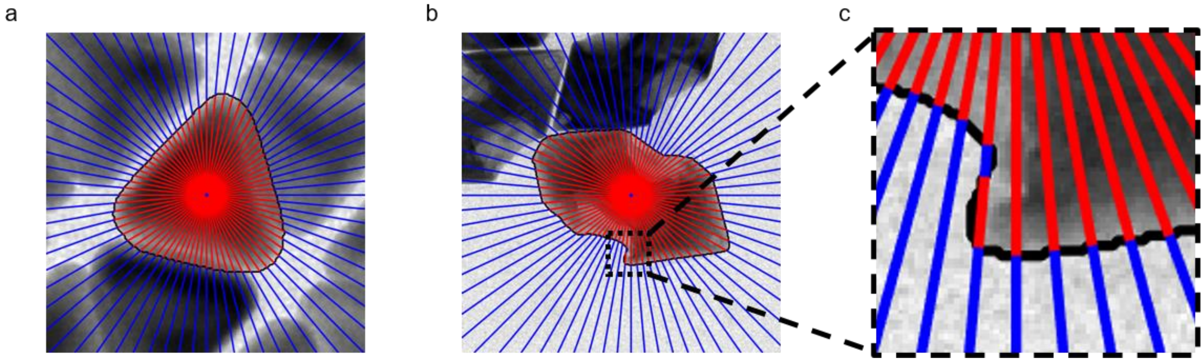

Figure S5. Measurement and modified definition of $d(\theta)$ for convex and concave shapes. (a) For convex shapes, “rays” starting from the nanoparticle centroid with orientation θ measure the centroid-to-contour distance $d(\theta)$ (red segments). (b–c) For concave shapes, rays can have multiple intercepts with the nanoparticle contour. Thus, we redefine $d(\theta)$ as the total distance of each ray traveled within the contour (sum of red segment lengths in each ray).

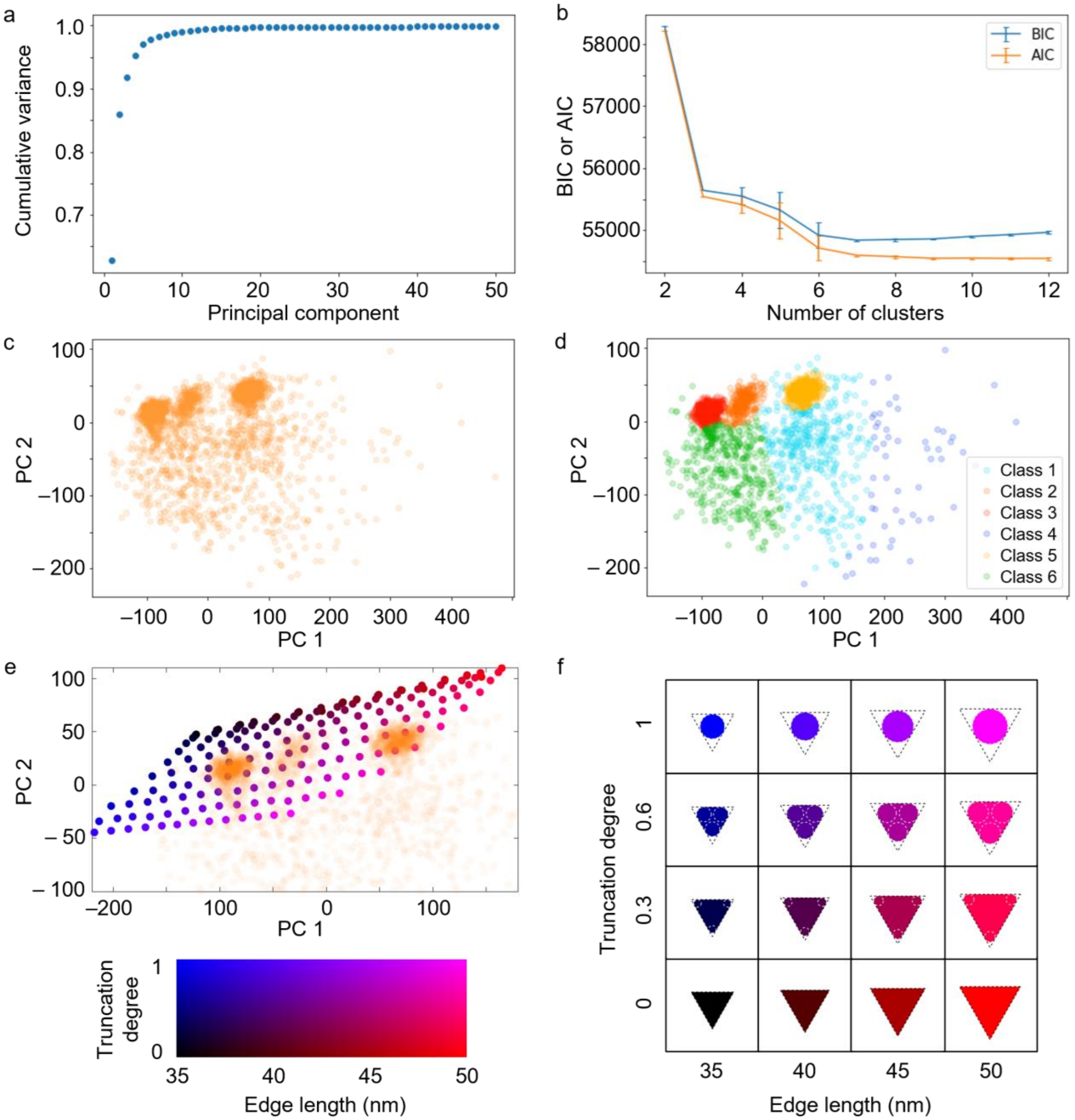

Figure S6. (a) Cumulative variance of principal components computed from the gold tetrahedral nanoparticle shape fingerprints. (b) BIC and AIC of GMMs with different number of clusters. Error bars represent the standard deviation of the information criterion of 20 randomly initialized GMMs. (c) Gold tetrahedral nanoparticle fingerprints projected onto their first two principal components. (d) Data points of panel (c) clustered by the GMM into six classes. (e) Theoretical equilateral triangular shapes with known edge length and truncation projected onto the first two principal components of the data set of experimental gold tetrahedral nanoparticles. Data points in gradient colors correspond to theoretical triangular shapes; data points in orange color represent the experimental nanoparticles. Colormap indicates the combination of edge length and degree of truncation. We use three circles moving along the bisectors of a sharp triangle and tangent to pairs of adjacent edges of the sharp triangle to construct theoretical truncated triangles (see panel (f)). We define the degree of truncation as $\frac{2R}{\tan(30°)\,L}$, where R and L are the radius of the best-fitted circle of the truncated tip and the edge length of the sharp triangle, respectively. Truncation degree = 0 corresponds to a sharp (untruncated) triangle with the given edge length; truncation degree = 1 corresponds to a triangle of the given edge length with its tips rounded such that it reduces to the inscribed circle. (f) Theoretical triangular shape examples with different size and truncation degree. White dashed lines indicate the best-fitted circles of the truncated tips used to construct the truncated triangles. Black dashed lines indicate the starting sharp triangles without any truncation. Colors are consistent with the colormap in panel (e).

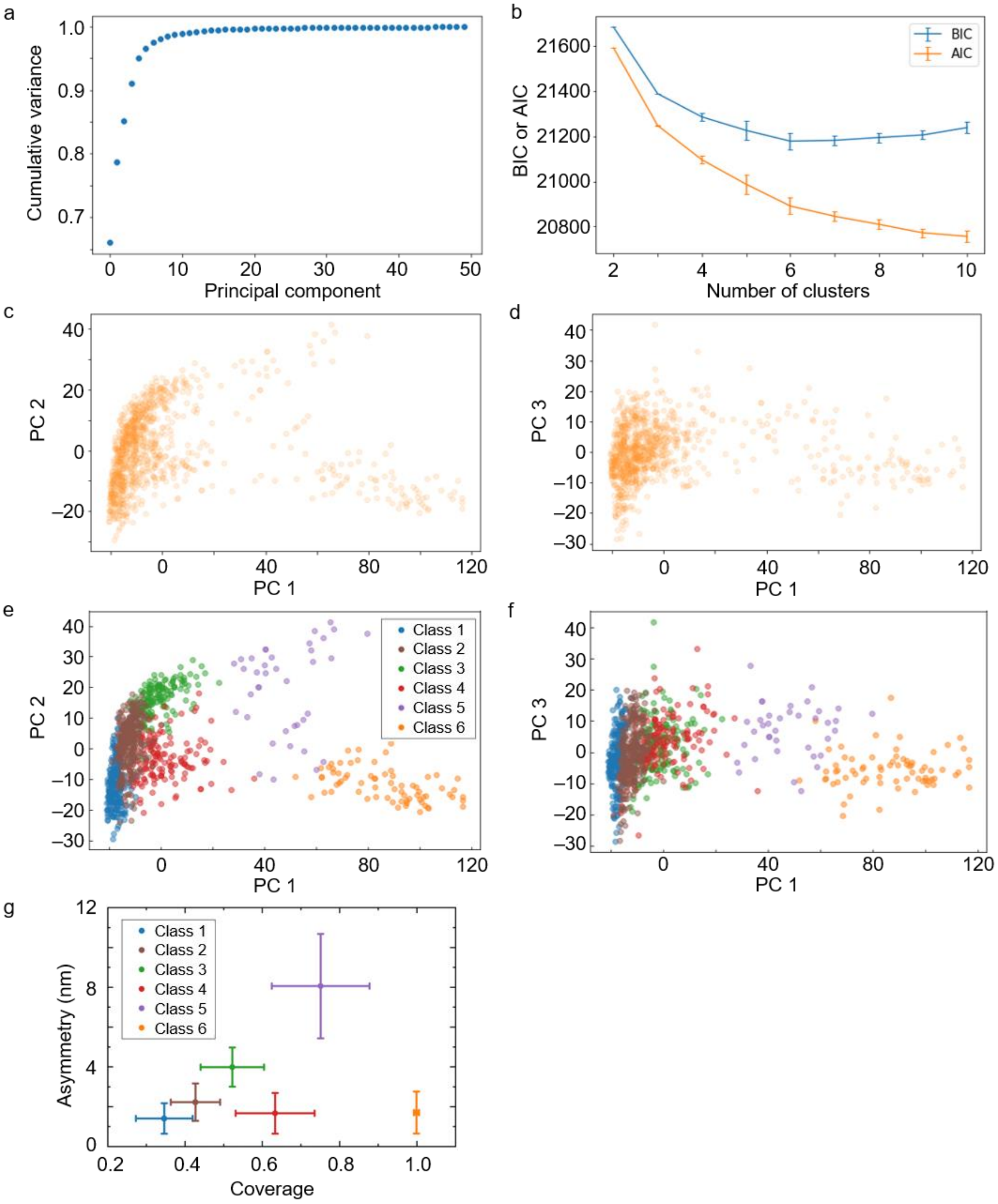

Figure S7. (a) Cumulative variance of principal components computed from the patchy nanoprism fingerprints. (b) BIC and AIC of GMMs with different numbers of clusters. Error bars represent the standard deviation of the information criterion of 20 randomly initialized GMMs. (c) Patchy nanoprism fingerprints projected onto their first and second principal components. (d) Patchy nanoprism fingerprints projected onto their first and third principal components. (e, f) Data points of panels (c) and (d) clustered by the GMM into five classes. (g) Asymmetry and coverage of shape fingerprints from each class predicted by GMM. Error bar indicates the standard deviation of the parameter among all fingerprints from each class.

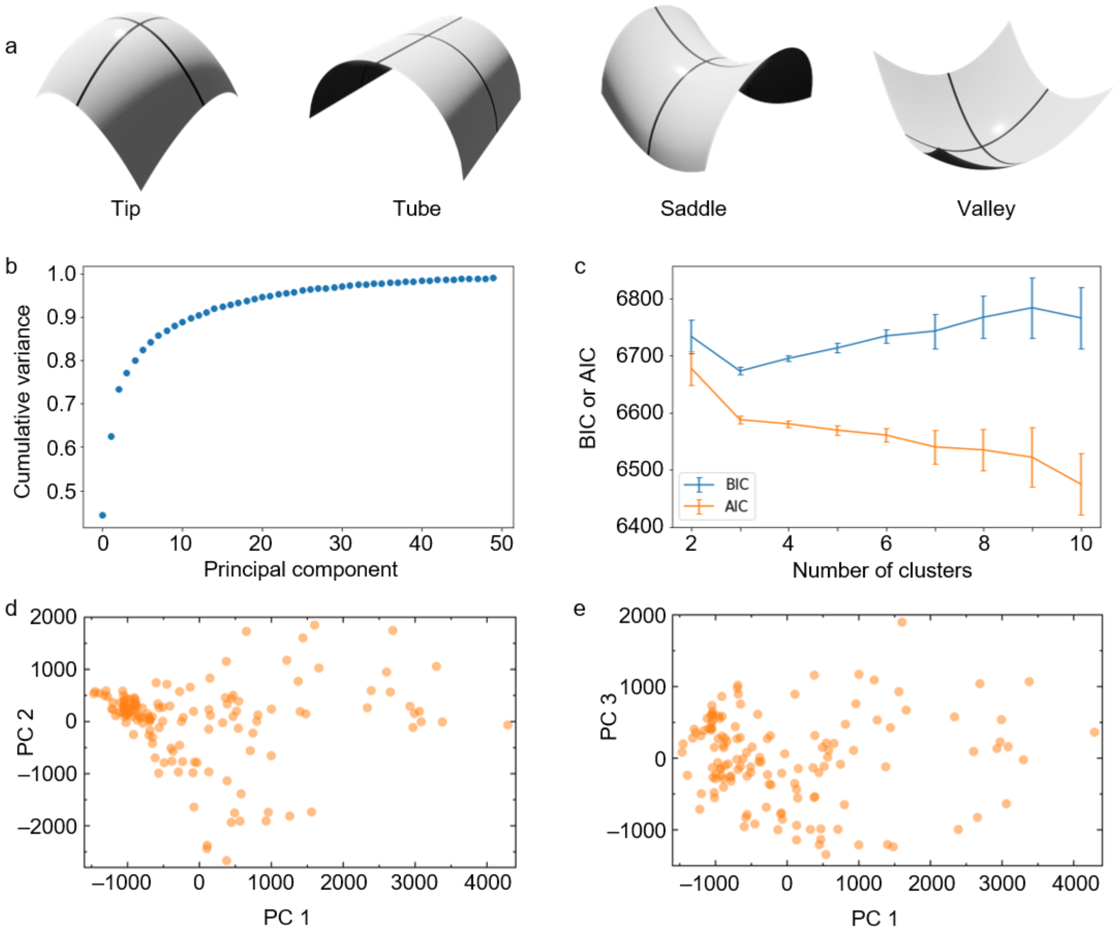

Figure S8. (a) Schematics showing the four surface curvature elements. (b) Cumulative variance of principal components computed from the polyamide membrane crumple 2D shape fingerprints. (c) BIC and AIC of GMMs with different numbers of clusters. Error bars represent the standard deviation of the information criterion of 20 randomly initialized GMMs. (d) Crumple shape fingerprints projected onto their first and second principal components. (e) Crumple shape fingerprints projected onto their first and third principal components.

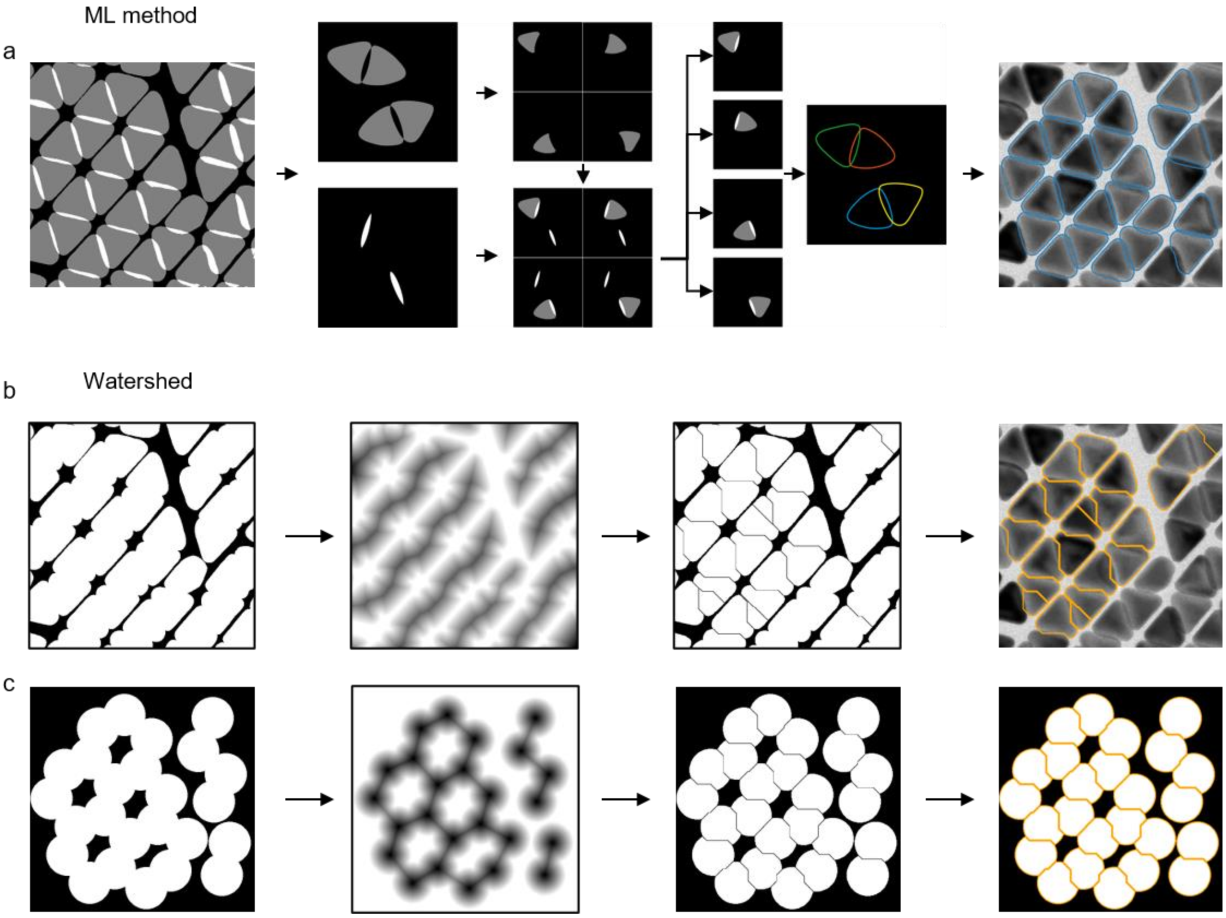

Figure S9. Separation of overlaying nanoparticles in TEM images with our machine learning method and with a conventional watershed algorithm. (a) Workflow showing the machine learning-based nanoparticle separation. Each isolated area in the non-overlaying channel (top) predicted by the U-Net is combined with the overlaying channel (bottom) to reconstruct the original overlaying nanoparticle contours. (b) Workflow showing the conventional watershed transform-based nanoparticle separation. The segmented image of overlaying nanoparticles (first column) goes through a distance transform (second column) and watershed transform (third column) to obtain an image with separated nanoparticles. Due to the particle shape and the degree of overlap, the separation performance is poor (fourth column). (c) Watershed transform applied to circular shapes yields the correct nanoparticle number (unlike the tetrahedral case in panel (b)), but does not identify the circular contours of overlapping disks.

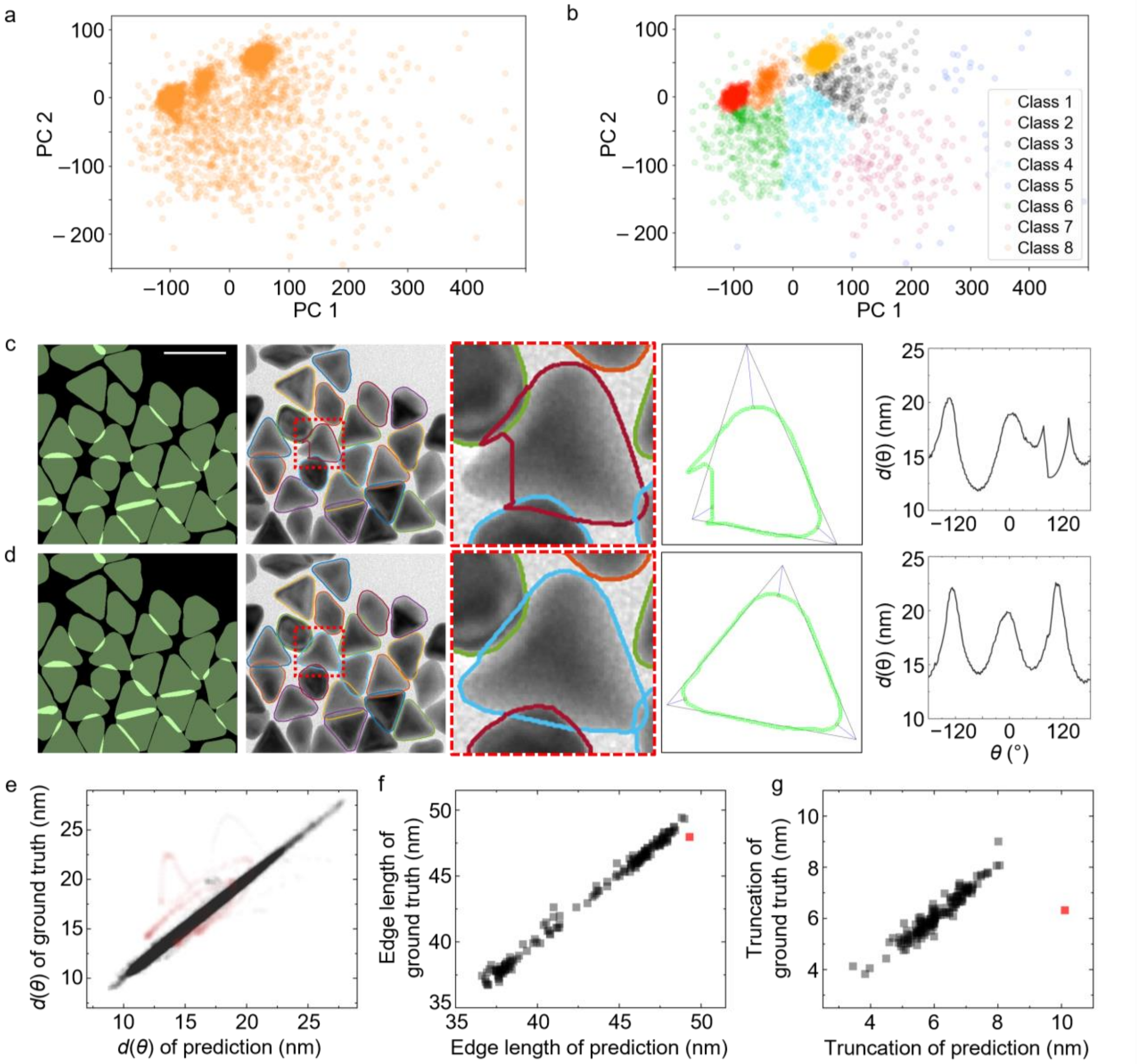

Figure S10. (a) Gold tetrahedral nanoparticle fingerprints projected onto their first two principal components without manually removing the failed detections. (b) Data points of panel (a) clustered by the GMM into eight classes. The three classes representing the tetrahedral nanoparticles (class 1, 2, and 8) are still consistent with the clustering result after removing the failed detections (Fig. S6d). (c) From left to right: U-Net prediction of experimental gold tetrahedral nanoparticle images with labeled ground truth; The particle contours reconstructed from the prediction denoted by different colors; Zoom-in view of a problematic contour delineated by the red dashed line box; The triangle fitting, edge length and truncation measurement of the particle; The fingerprint function of the particle. (d) From left to right: The manually labeled ground truth of the same image as in panel (c); The particle contours reconstructed from the ground truth denoted by different colors; Zoom-in view of the same particle as in (c) delineated by the red dashed line box; The triangle fitting, edge length and truncation measurement of the particle; The fingerprint function of the particle. (e–g) Scatter plots of (e) values from the fingerprint functions of ground truth and predicted reconstructions, (f) edge lengths and (g) truncations measured from ground truth and predicted reconstructions of gold tetrahedral nanoparticles. Red data points come from the particle delineated by the red dashed line box in (c) and (d); black data points come from the rest of particles. Scale bar: 50 nm.

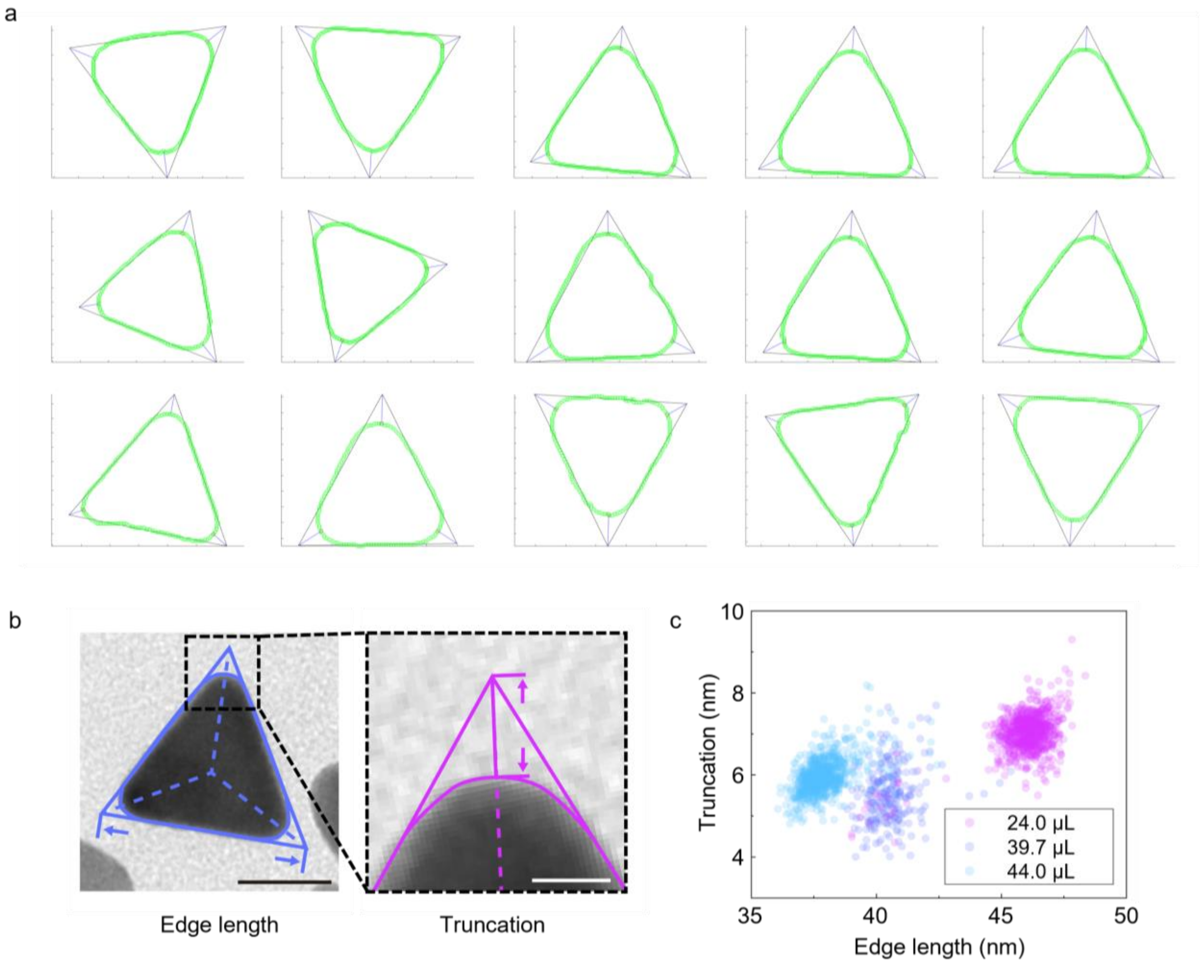

Figure S11. (a) Examples of the edge length and truncation measurement from tetrahedral nanoparticle contours using our algorithm. (b) Schematic showing our definitions of edge length and truncation. The colored dashed lines connect the tips of the fitted sharp triangle with its centroid to help the measurement of truncation. (c) Distribution of measured edge length and truncation of gold tetrahedral nanoparticles synthesized at various seed concentration. Scale bars: 25 nm in left panel of (b); 5 nm in right panel of (b).

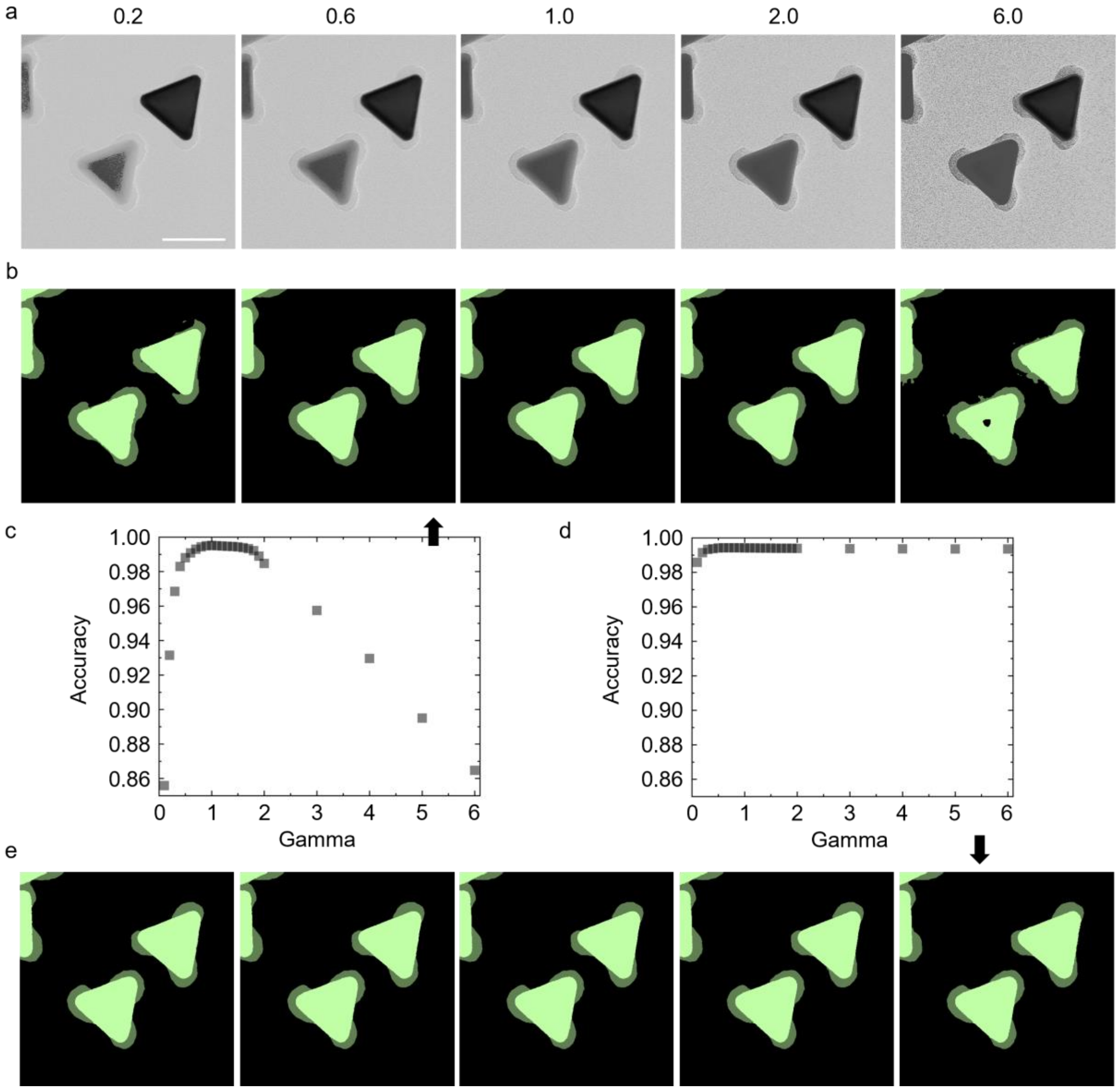

Figure S12. (a) Examples of experimental patchy nanoprism TEM images with contrast adjustments. The values on top of the images indicate the gamma values used for the contrast adjustments. (b) The predictions corresponding to panel (a) produced by the U-Net model used in this work. (c) The prediction accuracies as a function of contrast adjustment parameter gamma, produced by the U-Net model used in this work. (d) The prediction accuracies as a function of contrast adjustment parameter gamma, produced by another U-Net model trained with the contrast adjusted images included in the training dataset. (e) The predictions corresponding to panel (a) produced by the U-Net model trained with including the contrast adjusted images in the training dataset. Scale bar: 50 nm.

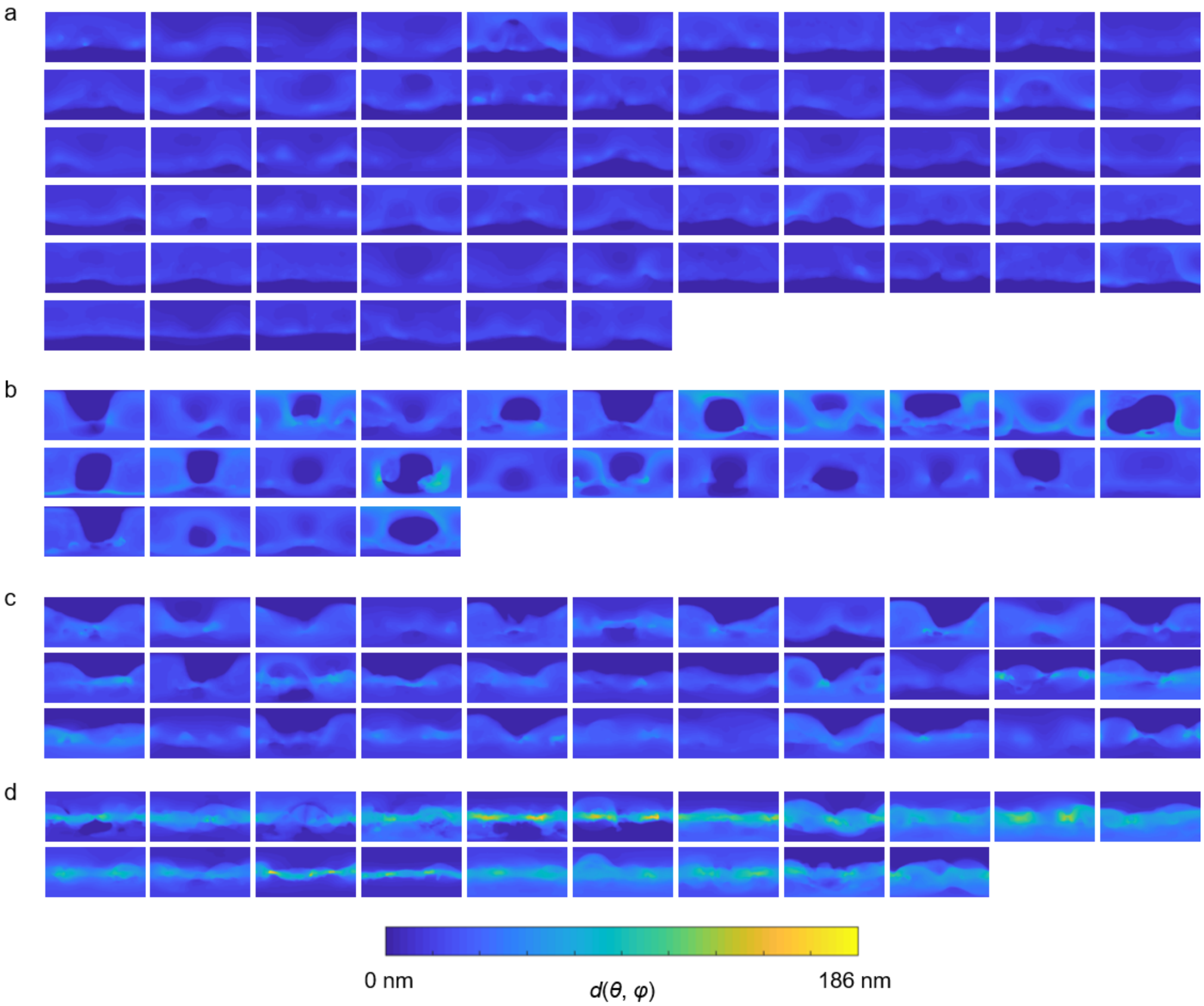

Figure S13. Shape fingerprints of 140 polyamide membrane crumples belonging to different classes identified by the GMM: (a) dome, (b) dimple, (c) lay, and (d) pancake.