Journal of Water and Environmental Nanotechnology (JWENT)

Supporting Information

for *J Water Environ Nanotechnol* 2026, 11(SI)

DOI: 10.22090/jwent.2025.2070247.1962

Effect of Zinc Doping on Structural, Optical, Magnetic, Effect of Zinc Doping on Structural, Optical, Magnetic, Photocatalytic, and Anti-Microbial Properties of ZrO_2 Nanoparticles via Machine Learning-Based SEM Image Analysis

Jeba Rengaswamy*1, Radhika Sathasivam2, Ascar Davix Xavier Durai3

1Department of Physics, Women's Christian College, Nagercoil, Tamilnadu, India - 629001.
2Department of Physics, Pioneer Kumaraswamy College, Nagercoil,Tamilnadu, India–629003.
3Department of Electronics and Communication Engineering, R.V.R. & J.C. College of Engineering, Guntur,

In the format provided by the authors and unedited

Fundamental Equations:

The crystallite size (D), dislocation density (δ), Stacking Fault (SF) and micro strain (ε) were calculated using equations 1, 2, 3 and 4 respectively.

$$D = \frac{0.9\lambda}{\beta \cos\theta} \quad (1)$$

where, 'λ' is the wavelength of the X-rays; 'θ' is the Bragg's diffraction angle, and 'β' is the full width at half maximum (FWHM) of the diffraction peaks (in radians).

$$\delta = \frac{1}{D^2} \quad (2)$$

$$SF = \frac{2\pi^2}{45\sqrt{3\tan\theta}} \quad (3)$$

$$\varepsilon = \frac{\beta \cos\theta}{4} \quad (4)$$

where β is the FWHM and θ is the diffraction angle.

The lattice parameters and the volume of unit cell are calculated using equations 5 and 6 respectively [26].

$$\frac{1}{d2} = \frac{h2+k2}{a2} + \frac{l2}{c2} \quad (5)$$

$$V = \frac{\sqrt{3}a^2c}{2} \quad (6)$$

Table 1. Lattice and geometrical parameters of synthesized pure and Zn-doped ZrO_2 nanoparticles

Sample detail	Lattice parameters				Geometrical parameters			
	a=b (Å)	C (Å)	Aspect ratio 'c/a'	Unit cell volume 'V' $(Å)^3$	Crystallite Size, D (nm)	Micro strain 'ε' (×10−3)	Dislocation density'δ' ($\times 10^{15}$ n/m2)	Stacking fault
Pure	3.59	5.27	1.47	58.82	29.74	1.26	1.23	0.48
Zn-1	3.604	5.13	1.42	57.57	11.40	2.83	7.69	0.49
Zn-2	3.61	5.07	1.40	57.22	11.87	2.82	7.1	0.49
Zn-3	3.60	5.11	1.41	57.35	11.91	1.74	7.05	0.49

Algorithm for the analysis of SEM images

Algorithm:

Step 1: Image Preprocessing and Calibration

1.1 Scale calibration:

- Extract scale bar information from image metadata or manual annotation
- Calculate pixels-to-microns conversion factor: γ = (physical length)/(pixel length)
- Store calibration factor for all subsequent measurements

1.2 Image normalization:

- Apply contrast-limited adaptive histogram equalization (CLAHE) to enhance features
- Normalize pixel intensities to [0, 1] range: I_norm = (I - min(I))/(max(I) - min(I))
- Resize image to YOLOv8-compatible dimensions (640 × 640) while maintaining aspect ratio with padding

1.3 Patch extraction for texture analysis:

- Divide original image into overlapping 224 × 224 patches P_j with stride 112
- Store spatial coordinates for each patch for reconstruction

Step 2: Instance Segmentation Using YOLOv8-Seg

2.1 Model initialization:

- Load pre-trained YOLOv8-seg model
- Configure confidence threshold (0.7) and IoU threshold (0.6)

2.2 Inference and segmentation:

- Process normalized image through YOLOv8 to obtain:
    - Bounding boxes B_i = [x_min, y_min, x_max, y_max, confidence]
    - Instance segmentation masks M_i for each detected particle
    - Class predictions C_i (if multiple particle types present)

2.3 Post-processing:

- Apply non-maximum suppression to eliminate overlapping detections
- Filter masks using confidence thresholding
- Convert mask coordinates to original image dimensions using inverse transformation

Step 3: Shape Analysis from Segmentation Masks

3.1 Morphological processing:

- Apply binary closing to fill small holes in masks: $M_i' = M_i \bullet K$, where K is 3×3 structuring element
- Smooth mask boundaries using morphological opening

3.2 Shape descriptor extraction for each mask M_i:

- Area: $A_i = \gamma^2 \times \Sigma$ pixels in M_i
- Perimeter: $P_i = \gamma \times$ chain code length of mask boundary
- Circularity: $C_i = (4\pi A_i)/(P_i)^2$
- Aspect ratio: AR_i = (Major axis length)/(Minor axis length)
- Solidity: $S_i = A_i$ / Area of convex hull
- Roundness: $R_i = (4A_i)/(\pi \times (\text{Major axis})^2)$

3.3 Particle classification by shape:

- Implement rule-based classification:
    - Spherical: $C_i > 0.9$, $AR_i < 1.2$
    - Rod-like: $AR_i > 3.0$
    - Cubes : $AR_i = 0.5$
    - Irregular: $C_i < 0.7$

Step 4: Agglomeration Analysis

4.1 Spatial clustering:

- Extract centroid coordinates for all detected particles: $C_i = [x_i, y_i]$
- Apply DBSCAN clustering with parameters $\varepsilon = 10$ pixels and min_samples = 3
- Identify cluster labels L_i for each particle

4.2 Agglomeration metrics:

- Agglomeration degree: $AD_k = N_k / A_k$ for each cluster k
- Primary particle ratio: PPR = (Number of single particles)/(Total particles)
- Cluster size distribution: Histogram of particles per cluster
- Inter-particle distance: Mean nearest-neighbor distance within clusters

4.3 Aggregate morphology:

- For each cluster, calculate:
    - Convex hull area
    - Fractal dimension using box-counting method
    - Compactness: (Area of cluster)/(Area of convex hull)

Step 5: Surface Topography Analysis Using ConvNeXt

5.1 Texture model initialization:

- Load pre-trained ConvNeXt-Base model
- Replace classifier head with custom head for texture classes (smooth, porous, rough, cracked)
- Load fine-tuned weights for SEM texture analysis

5.2 Patch-based texture classification:

- For each patch P_j:
    - Normalize using ImageNet statistics
    - Obtain texture class probabilities: T_j = ConvNeXt(P_j)
    - Assign dominant class: C_j = argmax(T_j)
- Reconstruct texture probability map for entire image

5.3 Topography quantification:

- Texture distribution: Percentage coverage of each texture class
- Spatial heterogeneity: Entropy of texture class distribution
- Feature size: Auto-correlation length for each texture class
- Surface roughness: Statistical analysis of intensity variations within texture classes