Competition 9: Intelligent Recognition and Analysis of Material Science Image Curves

1. Competition Background

Curves in materials science literature are one of the core carriers of material properties. For example, stress–strain curves contain information on mechanical properties such as strength, elongation, and work-hardening rate; cyclic voltammetry (CV) curves reflect electrochemical reaction kinetics; differential scanning calorimetry (DSC) curves characterize thermally induced phase transitions; and spectroscopic curves contain compositional or structural information. Statistics show that more than 70% of materials science publications contain curve images. However, in current research practice, curve data extraction still relies heavily on manual digitization software such as GetData, which suffers from low efficiency and high subjectivity (errors up to 1–3%), making it unsuitable for automated, high-precision numerical extraction and analysis. This competition aims to leverage artificial intelligence to overcome the technical bottleneck of curve image digitization and intelligent analysis, thereby promoting the automation and intelligent transformation of curve data acquisition in materials science.

2. Competition Application Scenario

Take the development of novel corrosion-resistant materials as an example: when analyzing a large volume of historical literature, researchers must extract cyclic voltammetry curve data from thousands of reports and derive key parameters such as redox potentials based on the curve's characteristic peaks and valleys. Traditional manual annotation not only introduces observational errors but also cannot meet the efficiency demands of rapidly growing data volumes. Conventional chart recognition tools struggle with complex cases such as segmented axes, overlapping curves, or legend occlusions, leading to significant declines in extraction success rates in practice. Developing new intelligent algorithms capable of accurately parsing curve images has become a common technical requirement across industries for data extraction. This competition requires participants to develop intelligent analysis algorithms that achieve automatic axis recognition, data point extraction, and batch processing for multiple curve types, producing structured outputs. The goal is to use AI technologies to break through the bottleneck of curve recognition and push materials science data acquisition toward automation and intelligence.

3. Competition Task

Teams must design algorithmic systems capable of parsing curve images, focusing on real-world challenges such as segmented axes, nonlinear scales, multiple curve intersections/adhesions, legend occlusions, and scanning noise. The algorithms should

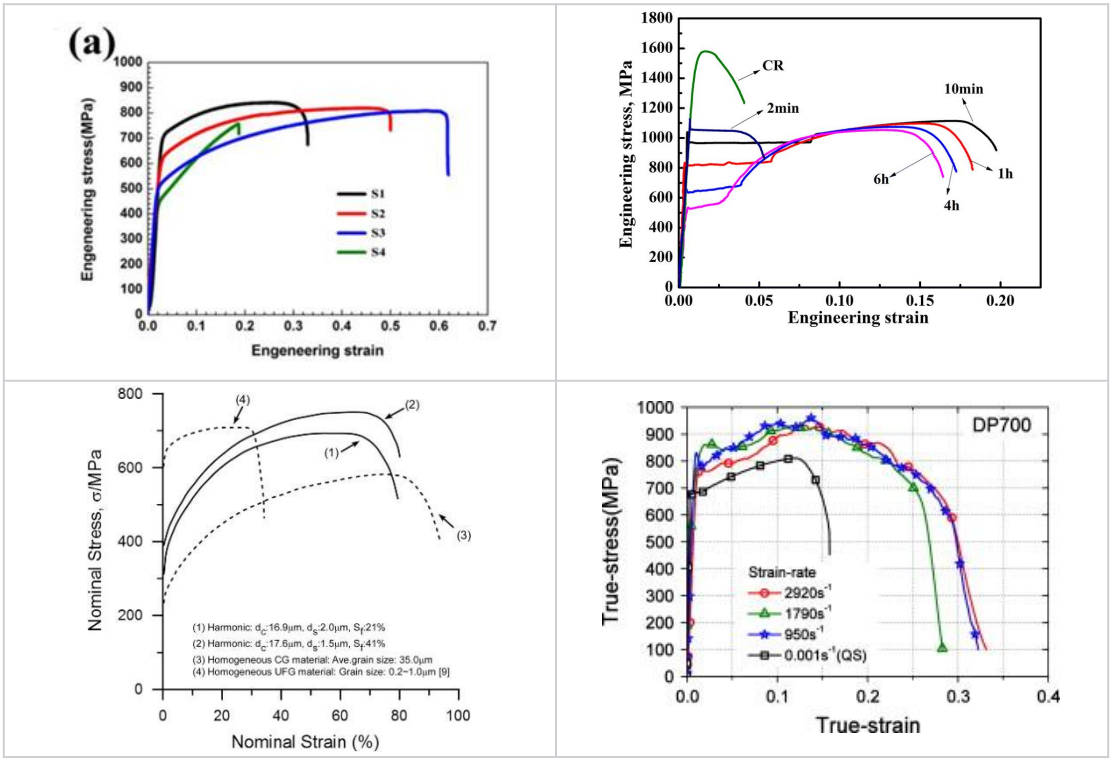
achieve pixel-level curve extraction, semantic axis parsing, and legend matching through multimodal feature fusion. Optionally, participants may incorporate materials knowledge constraints (e.g., monotonicity of stress–strain curves, periodicity of cyclic curves) to improve recognition reliability. Furthermore, the algorithms should overcome device-induced variations and be compatible with multiple types of materials characterization images (e.g., electrochemical polarization curves, differential scanning calorimetry curves), meeting the industrial-level requirement of cross-domain generalization.

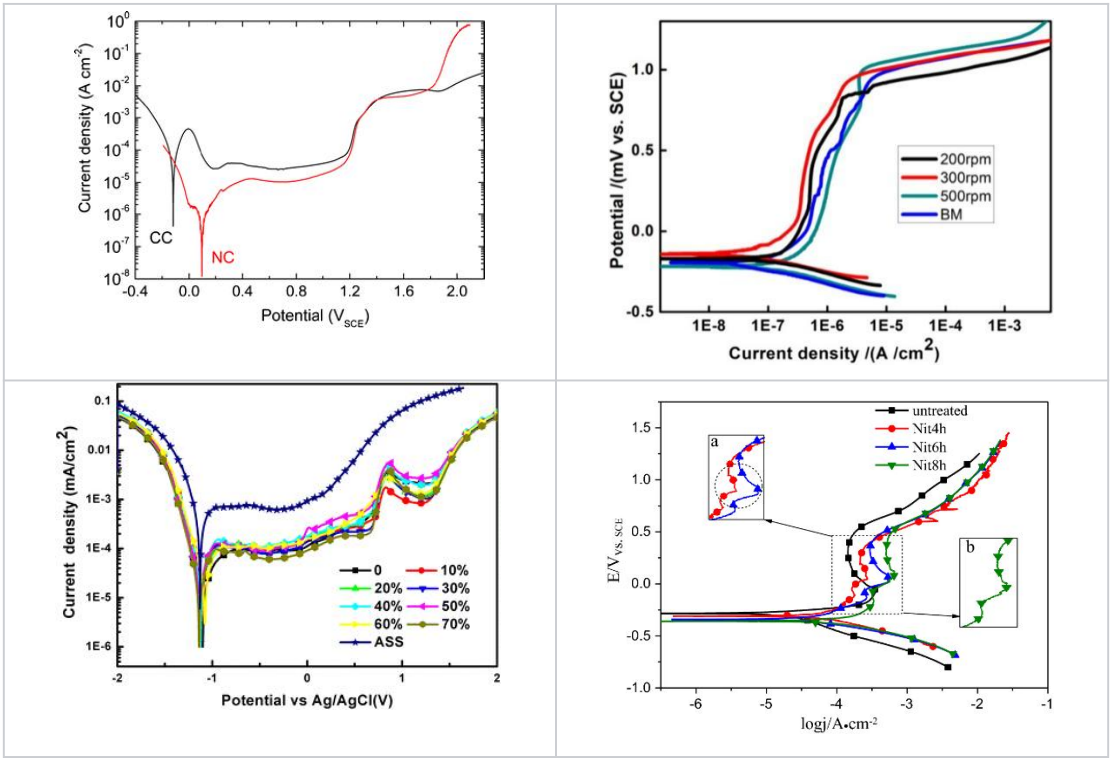
4. Dataset and Data Description

The dataset will be acquired through participant self-collection or by artificially generating simulated data. Teams may also extract curve images (with legends) from publicly available literature PDFs for training purposes.

Data format example:

(1) Input images (typically containing key elements such as coordinate axes, curves, and legends):





(2) Standard Extraction Results

The output should mainly include the following three categories of structured information: (1) The numerical sequence corresponding to each curve. (2) Extracted image metadata, including the physical quantities and units of the coordinate axes, and the mapping between curves and legends.

(3) Output Example

The final output should be an Excel file. The results below correspond to the extraction of image (a), with the data columns, from left to right, being: figure_name, figure_index, figure_title, x-label, y-label, sample, point_coordinates, and note (other textual annotations in the image).

figure_name (id)	figure_index	figure_title	x-label	y-label	sample	point_coordinates	note
stress_strain_curve_test.png	(a)		Engeneering strain	Engenerring stress(MPa)	S1	0.00003, 12.22707 0.00277, 27.94760	
stress_strain_curve_test.png	(a)		Engeneering strain	Engenerring stress(MPa)	S2	0.00003, 12.22707 -0.00127, 35.80786	
stress_strain_curve_test.png	(a)		Engeneering strain	Engenerring stress(MPa)	S3	0.00001, 4.36681 0.00276, 25.32751	
stress_strain_curve_test.png	(a)		Engeneering strain	Engenerring stress(MPa)	S4	-0.00133, 6.98690 0.00279, 35.80786	

1) figure_name: Use the original image filename; keep the name unchanged.

2) figure_index: For multi-subplot figures, label the corresponding subplot (e.g., (a), (b)); leave blank for a single-curve figure.

3) figure_title: Extract the main title at the top of the image; leave blank if none.

4) Label data points sequentially; pay special attention at curve inflection points; ensure at least 128 sampled points per curve.

5) Strictly follow the ascending direction of the x-axis (left to right); for oscillatory curves, sample according to the main trend.

6) Some images use logarithmic axes; distinguish them from linear axes, and convert all sampled values to the linear scale in the final output.

key issues to be addressed

Key Issue	Importance
Segmented axes	★★
Nonlinear axes	★★★
Multiple axes	★
Curve occlusion	★★★
Overlapping curves	★★★
Extraction of key points in line charts	★★★
Step-wise value sampling in curve charts	★★★
Curve-to-legend correspondence	★★★
Inclusion of extreme points	★★
Alignment of sampled points with legends	★★★
Low resolution of some images	★

5. Algorithm Design Requirements

5.1 Participants are encouraged to adopt deep learning–based frameworks to build vision-based object detection models with end-to-end processing capability. It is recommended to integrate the strengths of mainstream detection networks such as YOLO and Faster R-CNN, while incorporating global modeling methods like Transformer, DETR, or ViT for cross-modal feature fusion, in order to enhance multi-task analysis of axes, scales, text, and curves in images.

5.2 Participants are encouraged to propose innovative fusion strategies on top of existing methods, such as multi-scale feature extraction enhanced by attention mechanisms, self-supervised pretraining for auxiliary localization, or prior structures guided by domain knowledge. The aim is to further improve detection accuracy, generalization, and robustness. The use of encapsulated commercial software APIs (e.g., WebPlotDigitizer, PlotDigitize) for direct image data extraction is strictly prohibited; all algorithmic modules must be independently developed by the participating teams or reasonably extended from open-source frameworks.

5.3 Algorithm design should balance practicality and scalability, featuring a well-structured modular design that can be flexibly deployed on devices with different computational resources (e.g., GPU servers, edge computing terminals). It should support batch processing and multithreaded acceleration. When handling high-resolution images or datasets at the scale of thousands of images, the system should maintain stable performance and controllable memory usage, while providing fault-tolerance mechanisms and reliable data output.

6. Performance Metrics Requirements

This competition takes typical curve images of steel materials as the benchmark test objects and establishes a standardized performance evaluation system. The preliminary and semifinal stages each include 50 annotated images (such as stress–strain curves, polarization curves, and spectrograms). Performance evaluation mainly includes three aspects: using Dynamic Time Warping (DTW) distance to measure curve shape similarity, evaluating semantic accuracy with the F1-score, and measuring runtime efficiency under a GPU environment. Participating teams must submit complete training code and model parameters to support reproducibility verification. The ultimate goal of the competition is to develop an intelligent system that surpasses the precision of professional manual digitization, and by introducing small-sample learning methods enhanced with physical constraints, to significantly improve the efficiency of materials data processing and provide core support for high-throughput materials research and database construction. The required system performance is a curve value extraction error of less than 3% and a legend matching accuracy higher than 97%.

7. Functional Requirements

The system must support input images in multiple formats (such as PNG and JPG). The input images typically contain key elements such as coordinate axes, curves, and legends. The output results should include the following three categories of structured information: (1) the numerical sequence corresponding to each curve (in CSV format); (2) the extracted image metadata, including the physical quantities and units of the axes as well as the mapping between curves and legends.

8. Development Environment

8.1 Software Environment

(1) The software system used during development is not restricted (Windows, Linux, Mac, etc.), but the final code must be verified to run successfully on Linux. Development tools (IDEs) are not limited.

(2) Python should be the primary programming language. If other languages are used,

participants must provide proper encapsulation and use Python as the glue language to implement the relevant interfaces. The required versions of supporting libraries and frameworks should be clearly specified.

(3) It is recommended to use PyTorch as the deep learning framework to improve version compatibility.

8.2 Hardware Environment

The hardware environment during development is not restricted. However, the final code must at least be able to perform data extraction on a standard machine (32 cores, 24 GB GPU memory, Nvidia 4090). The image processing speed on the standard machine will be used as the basis for evaluating model inference efficiency.

9. Evaluation Criteria

The final offline score consists of objective evaluation (85%) and subjective evaluation (15%). The objective score is calculated based on the results of the manually annotated test set provided by the organizers, including: curve extraction accuracy (measured by DTW, 70%), recognition accuracy (20%), and model inference efficiency (a weighted score combining GPU memory usage and runtime, 10%). The subjective score is determined by materials science experts, who comprehensively evaluate two dimensions: algorithmic innovativeness (50%) and practical engineering value (50%). The total score is out of 100 points, with other solutions normalized proportionally. (For the second and third prize at the national level, the scores do not involve the 30% subjective evaluation component.)

10. Problem-Solving Approach

The curve images are first processed with adaptive filtering, color space analysis, dynamic threshold segmentation, and perspective transformation to achieve denoising, correction, and enhancement, ensuring image clarity and structural integrity. A two-stage coordinate parsing system is then employed: Hough transform and connected component analysis are used to locate axes and scales, followed by improved OCR for numeric recognition. Combined with semantic segmentation and trend prediction techniques, the system extracts curve data and automatically converts it into physical quantities, supporting multiple coordinate systems such as linear and logarithmic. Neural networks with transfer learning are adopted, incorporating style transfer and physical constraints to improve performance under small-sample conditions. In the deployment stage, a parallel processing pipeline and error correction mechanisms are developed to enable efficient parsing of multi-format images. The focus is placed on enhancing robustness in axis recognition and reconstructing occluded data, while integrating materials science domain knowledge and establishing a multi-dimensional evaluation system to ensure the practical engineering value

of the extracted results.

11. References

[1] Cliche, M., Rosenberg, D., Madeka, D., & Yee, C. (2017). Scatteract: Automated extraction of data from scatter plots. In Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18–22, 2017, Proceedings, Part I (pp. 135–150). Springer International Publishing.

[2] Siegel, N., Horvitz, Z., Levin, R., Divvala, S., & Farhadi, A. (2016). Figureseer: Parsing result-figures in research papers. In Computer Vision – ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII (pp. 664–680). Springer International Publishing.

[3] Luo, J., Li, Z., Wang, J., & Lin, C. Y. (2021). ChartOCR: Data extraction from chart images via a deep hybrid framework. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 1917–1925).

[4] Drevon, D., Fursa, S. R., & Malcolm, A. L. (2017). Intercoder reliability and validity of WebPlotDigitizer in extracting graphed data. Behavior Modification, 41(2), 323–339.

12. Submission Requirements

(1) Runnable Python source code and result files. Participants must perform predictions on the test set provided by the organizers. The code should include the main structure of the algorithm and be executable directly to ensure reproducibility of the results. The final code must generate a file named "extractor.xlsx", which will be used for unified evaluation. The data format of extractor.xlsx should follow the attachment.

(2) A report document in Word format: "Algorithm Description.docx", containing an introduction to the main principles of the algorithm, explanation of innovations, code logic (training, inference, etc.), references, and no fewer than 3000 words.

(3) A supporting folder named "support", including parameter files, weight files, and other materials required to run the code.

(4) For the curve image dataset, the extracted results must be saved as "extractor.xlsx" and submitted in a compressed ZIP format.

13. Contact Information

Competition Q&A group on QQ: 982830277

Email: shaohan.tian@hotmail.com

Official registration website: www.aicomp.cn

Appendix: Competition Process and Award Settings

(1) Registration Stage

Participants complete registration on the official competition website, submit personal or team information, and obtain the download link for the preliminary dataset.

(2) Preliminary Stage

Participants independently construct training datasets by collecting data, and use the preliminary test set provided by the organizers for validation and debugging of their methods. During the preliminary stage, there is no limit on the number of submissions per day, and the leaderboard will be updated regularly.

(3) Semifinal (Provincial) Stage

After the preliminary stage ends, the semifinal stage begins, with access to the semifinal dataset download link. Only teams that submitted valid results in the preliminary stage are eligible to enter the semifinal. During the semifinal, participants use the dataset provided by the organizers for model debugging and submit inference results on the semifinal test set. The semifinal lasts for 3 days, and each team may submit only once per day.

(4) Semifinal (Provincial) Results Announcement

The semifinal results will be announced on the official competition website. The number of teams entering the semifinal serves as the basis for awarding, and the proportion of awards will not exceed the provincial-level competition quota. First, second, and third prizes will be awarded in the semifinal (with provincial award certificates). Submissions whose algorithm performance falls below the baseline reference score provided by the organizers will be deemed invalid and will not be awarded. Winners of the first and second prizes in the semifinal advance to the national final.

(5) Final (National) Stage

1) Online Evaluation

Based on the semifinal leaderboard results, and with the number of finalist teams as the award basis, the proportion of awards will not exceed the national-level competition quota. A list of first-prize candidates and the winners of the second and third prizes (with national award certificates) will be announced.

2) Submission of Final Works

Candidate teams for the national first prize must submit technical documentation, algorithm code and model files, demonstration videos, and supplementary materials within the specified timeframe. After the submission deadline, no modifications or additions will be accepted.

3) Final Review Stage

A professional review panel will reproduce and audit the results of the national first-prize candidate teams. If questions arise during the review process, participants may be required to provide clarification.

4) Offline Grand Final

Candidate teams for the national first prize must submit the finalized technical documentation, algorithm code and model files, demonstration videos, and supplementary materials within the specified timeframe, and participate in the offline grand final defense. The winners and ranking of the national first prize will be determined based on both algorithm performance scores and defense scores. Failure to attend the offline defense will be considered as forfeiture of the award. National first-prize winners will receive honorary certificates.