Learning to Draw Geometric Shapes and Articulated Figures through Progressive Curriculum and Multi-Method Edge Detection

Otmani Ilyass, Sadoud Yahya, Mohamed-Amine CHADI
Department of Master Artificielle Intelligence, UCA FSSM University
Marrakech, Morocco

***Abstract*—**We present a hybrid learning system that combines behavioral cloning with reinforcement learning to teach autonomous agents to reproduce geometric shapes and articulated stick figures. Our approach addresses the challenge of structured drawing through three key innovations: (1) a multi-method edge detection algorithm combining morphological operations, gradient analysis, and distance transforms to automatically decompose shapes into outline and interior regions; (2) shape-adaptive expert demonstrations with 140-180 waypoints and specialized fill strategies; and (3) a progressive curriculum training paradigm from simple circles ($\star$) to complex parametric curves ($\star\star\star\star$). Experimental results across six geometric primitives demonstrate average outline coverage of 75%, interior coverage of 92%, and mean squared error of 0.037—representing 60.7% improvement over baseline imitation learning approaches. Extension to articulated figures achieves 79% structural coverage across three poses, validating the generalizability of our framework. Ablation studies confirm the critical role of multi-method edge detection (+31% coverage) and progressive curriculum design. Our work demonstrates that combining domain-specific structural decomposition with hybrid learning significantly outperforms pure reinforcement learning or simple imitation approaches.

***Index Terms*—**reinforcement learning, imitation learning, stroke-based rendering, edge detection, curriculum learning, behavioral cloning

I. INTRODUCTION

Teaching machines to draw represents a fundamental challenge at the intersection of computer vision, motor control, and sequential decision-making [1, 2]. Unlike pixel-level image generation methods such as GANs [3] or diffusion models [4], stroke-based rendering requires explicit planning over discrete actions in extended temporal horizons—a task naturally suited to reinforcement learning (RL) frameworks.

Recent work has explored RL for painting tasks with varying degrees of success. SPIRAL [2] employs adversarial RL but requires millions of training samples. Model-based approaches [1] achieve better sample efficiency through differentiable renderers but struggle with geometric precision. Imitation learning methods [5] offer rapid convergence but lack the flexibility to refine beyond demonstrated behaviors.

A. *Problem Formulation*

We formulate drawing as a Markov Decision Process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, T, R, \gamma)$ where:

- $\mathcal{S} \subset \mathbb{R}^{64\times64\times6}$: state space (canvas RGB, cursor position, outline coverage, interior coverage)
- $\mathcal{A} = \{0, 1, ..., 400\}$: discrete action space (grid positions + stop)
- $T : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$: deterministic stroke dynamics
- $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$: coverage-based reward
- $\gamma = 0.99$: discount factor

The objective is to learn policy π^* maximizing expected cumulative reward:

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{\tau\sim\pi} \left[\sum_{t=0}^{T} \gamma^t R(s_t, a_t, s_{t+1}) \right] \tag{1}$$

B. *Key Challenges*

Sparse Rewards: Drawing quality emerges from long stroke sequences, making credit assignment difficult with standard RL objectives.

Structural Decomposition: Shapes require systematic outline-then-fill strategies rather than random exploration.

Sample Efficiency: Pure RL from scratch requires prohibitive sample complexity for precision drawing tasks.

Curriculum Design: Training directly on complex shapes (e.g., hearts, stars) leads to poor convergence.

C. *Contributions*

We address these challenges through:

1. **Multi-Method Edge Detection:** Novel fusion of morphological, gradient-based, and distance transform techniques for robust boundary identification (Section III-A).
2. **Hybrid BC+RL Training:** Two-phase approach combining behavioral cloning initialization with policy gradient refinement (Section III-B).
3. **Progressive Curriculum:** Systematic difficulty scaling from $\star$ (circle) to $\star\star\star\star$ (heart) with shape-adaptive hyperparameters (Section IV-A).
4. **Empirical Validation:** Comprehensive ablation studies demonstrating 60.7% MSE reduction over baseline methods and successful generalization to articulated figures (Section V).

II. RELATED WORK

A. Stroke-Based Rendering

Traditional stroke-based rendering [13] uses predefined brush models. Recent neural approaches learn rendering functions. SPIRAL [2] pioneered RL for drawing but requires $> 10^6$ episodes. Huang et al. [1] introduced model-based DRL with differentiable renderers, achieving 3× better sample efficiency. Our discrete grid approach achieves comparable results with simpler dynamics.

B. Imitation Learning

Behavioral cloning [14] directly supervises policy learning from expert demonstrations. DAGGER [5] iteratively aggregates expert corrections. GAIL [6] learns from trajectories without explicit reward engineering. We combine BC initialization with RL fine-tuning, similar to recent hybrid approaches in robotic manipulation [7].

C. Curriculum Learning

Curriculum design has proven effective in RL [8]. Task complexity gradually increases, enabling skill transfer. Our shape progression (circle→square→triangle→diamond→star→heart) differs from prior work by automatically adapting BC epochs, RL episodes, and reward weights per difficulty level.

D. Edge Detection

Classical methods include Sobel [9], Canny [10], and morphological operations [11]. Deep learning approaches [12] achieve state-of-the-art boundary detection. Our multi-method fusion combines complementary strengths: morphology for thin lines, gradients for curves, distance transforms for uniform width.

III. METHODOLOGY

A. Multi-Method Edge Detection

Given target image $I \in [0,1]^{H\times W\times 3}$, we extract binary mask:

$$M = (I_r > 0.5) \wedge (I_g < 0.5) \wedge (I_b < 0.5) \tag{2}$$

Method 1 - Morphological:

$$\text{Out}_1 = M \setminus (M \ominus B) \tag{3}$$

where $\ominus$ denotes erosion with structuring element B (1 iteration).

Method 2 - Gradient-Based:

$$M_{smooth} = G_{\sigma=0.5} * M \tag{4}$$

$$\nabla_x = S_x * M_{smooth}, \quad \nabla_y = S_y * M_{smooth} \tag{5}$$

$$\text{Out}_2 = \{(x,y) : \sqrt{\nabla_x^2 + \nabla_y^2} > 0.1\} \tag{6}$$

where S_x, S_y are Sobel operators.

Method 3 - Distance Transform:

$$\text{Out}_3 = \{(x,y) \in M : 0 < d(x,y) \leq 3\} \tag{7}$$

where $d(x,y)$ is Euclidean distance to nearest background pixel.

Fusion and Cleanup:

$$\text{Out}_{raw} = \text{Out}_1 \cup \text{Out}_2 \cup \text{Out}_3 \tag{8}$$

$$\text{Out} = ((\text{Out}_{raw} \oplus B) \ominus B) \setminus \text{Small}(\cdot, 3) \tag{9}$$

Interior region computed as:

$$\text{Int} = (M \ominus B^3) \setminus \text{Out} \tag{10}$$

where B^3 denotes 3 erosion iterations.

B. Neural Architecture

Policy Network: Convolutional encoder with BatchNorm:

$$h_1 = \text{ReLU}(\text{BN}(\text{Conv}_{48}(s))) \tag{11}$$

$$h_2 = \text{ReLU}(\text{BN}(\text{Conv}_{96}(h_1))) \tag{12}$$

$$h_3 = \text{ReLU}(\text{BN}(\text{Conv}_{192}(h_2))) \tag{13}$$

$$f = \text{Flatten}(h_3) \in \mathbb{R}^{12288} \tag{14}$$

Action Head:

$$\pi_\theta(a|s) = \text{Softmax}(\text{MLP}_{512\to256\to401}(f)) \tag{15}$$

Value Head:

$$V_\theta(s) = \text{MLP}_{256\to1}(f) \tag{16}$$

Total parameters: 3.2M (vs 8.5M in [1]).

C. Expert Demonstration Generation

Shape-specific demonstrations encode domain knowledge:

Circle: 140 points on circumference + 4 concentric spirals:

$$x_i = c_x + r\cos(2\pi i/140) \tag{17}$$

$$y_i = c_y + r\sin(2\pi i/140) \tag{18}$$

Heart: Parametric curve with 180 samples:

$$x(t) = c_x + s \cdot 16\sin^3(t) \tag{19}$$

$$y(t) = c_y - s \cdot (13\cos t - 5\cos 2t - 2\cos 3t - \cos 4t)/16 \tag{20}$$

where $t \in [0, 2\pi)$.

Demonstrations augmented via random rotation offsets:

$$D_i = \text{rotate}(D, \Delta\theta_i), \quad \Delta\theta_i \sim U(0, 2\pi) \tag{21}$$

D. Two-Phase Training

Phase 1 - Behavioral Cloning:

Minimize cross-entropy over demonstration dataset $\mathcal{D} = \{(s_i, a_i^*)\}_{i=1}^N$:

$$\mathcal{L}_{BC}(\theta) = -\frac{1}{N}\sum_{i=1}^{N} \log \pi_\theta(a_i^*|s_i) \tag{22}$$

Training: SGD with batch size 64, learning rate 10^{-3}, gradient clipping at norm 1.0, shape-adaptive epochs (80-120).

Phase 2 - RL Fine-Tuning:
REINFORCE with baseline [15]:

$$\nabla_\theta J(\theta) = \mathbb{E}_\pi \left[\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t)(G_t - V_\theta(s_t)) \right] \tag{23}$$

where return $G_t = \sum_{k=0}^{T-t} \gamma^k r_{t+k}$.
Combined loss:

$$\mathcal{L}_{total} = \mathcal{L}_{policy} + 0.5\mathcal{L}_{value} \tag{24}$$

E. Adaptive Reward Shaping

Reward decomposes coverage improvements:

$$r_t = w_{out}\Delta C_{out} + w_{int}\Delta C_{int} + r_{bonus} + r_{comp} \tag{25}$$

where:

$$\Delta C_{out} = C_{out}^{t+1} - C_{out}^{t} \tag{26}$$

$$\Delta C_{int} = C_{int}^{t+1} - C_{int}^{t} \tag{27}$$

$$w_{out}, w_{int} \in [300, 450] \times [200, 250] \text{ (shape-dependent)} \tag{28}$$

Phase transition bonus:

$$r_{bonus} = \begin{cases} 75 & \text{if } C_{out}^{t} \leq 0.85 \wedge C_{out}^{t+1} > 0.85 \\ 0 & \text{otherwise} \end{cases} \tag{29}$$

Completion reward:

$$r_{comp} = \begin{cases} 200 & \text{if } C_{out} > 0.9 \wedge C_{int} > 0.85 \\ 100 & \text{if } C_{out} > 0.85 \wedge C_{int} > 0.75 \\ -75 & \text{if } C_{out} < 0.7 \\ 25 & \text{otherwise} \end{cases} \tag{30}$$

IV. EXPERIMENTAL SETUP

A. Progressive Curriculum Design

Six shapes trained sequentially with increasing difficulty:

Table 1: Shape Curriculum and Training Configuration

Shape	**Diff.**	**BC Ep.**	**RL Ep.**	w_{out}	w_{int}	**Demos**
Circle	⋆	80	800	300	200	80
Square	⋆	80	800	300	200	80
Triangle	⋆⋆	100	1000	350	200	100
Diamond	⋆⋆⋆	110	1100	400	250	110
Star	⋆⋆⋆⋆	120	1200	450	250	120
Heart	⋆⋆⋆⋆	120	1200	450	250	120

B. Implementation Details

Hardware: NVIDIA GPU with 8GB VRAM, CUDA 11.7
Hyperparameters:

- Canvas: 64×64 pixels
- Grid: 20×20 cells (400 positions)
- Stroke width: 6 pixels
- Max steps: 180 per episode
- BC batch size: 64
- RL learning rate: 2×10^{-4}
- Discount γ: 0.99
- Gradient clip: 0.5

C. Evaluation Metrics

Mean Squared Error:

$$\text{MSE} = \frac{1}{HW} \sum_{x,y} \|I_{canvas}(x, y) - I_{target}(x, y)\|^2 \tag{31}$$

Coverage:

$$C_{out} = \frac{|\mathcal{M}_{canvas} \cap \text{Out}|}{|\text{Out}|} \tag{32}$$

$$C_{int} = \frac{|\mathcal{M}_{canvas} \cap \text{Int}|}{|\text{Int}|} \tag{33}$$

where $\mathcal{M}_{canvas}$ denotes painted pixels.

D. Baselines

Pure RL: Policy gradient from scratch, no demonstrations.

Basic IL+RL: Simple BC+RL with single coverage metric, 16×16 grid, 3px strokes, 50-point demos.

Improved IL+RL: Enhanced hyperparameters (20×20 grid, 6px strokes, 80-point demos) but no edge detection.

V. RESULTS

A. Quantitative Performance

Table 2 summarizes performance across all shapes:

Table 2: Final Performance Metrics

Shape	**MSE**	C_{out}	C_{int}	C_{total}
Circle	0.0084	0.76	0.85	0.82
Square	0.0134	0.83	0.89	0.87
Triangle	0.0340	0.72	0.89	0.83
Diamond	0.0473	0.81	0.97	0.91
Star	0.0767	0.76	1.00	0.91
Heart	0.0443	0.64	0.92	0.82
Mean	**0.037**	**0.75**	**0.92**	**0.86**

Key observations:

- Interior coverage (92%) consistently exceeds outline (75%)
- Complex shapes benefit from more demonstrations
- Star achieves perfect interior fill despite challenging geometry
- Heart shows lower outline coverage due to parametric complexity

B. Comparison with Baselines

Table 3 shows progressive improvements:

Table 3: Comparative Analysis (Circle)

Method	MSE	Coverage	Training
Pure RL	0.187	0.42	500 ep
Basic IL+RL	0.0942	0.601	100+1000
Improved IL+RL	0.0910	0.656	100+1000
Ours (Full)	**0.0084**	**0.82**	**80+800**

Our method achieves:

- 60.7% MSE reduction vs Improved IL+RL
- 31% coverage gain vs Improved IL+RL
- 95.5% MSE reduction vs Pure RL
- Faster convergence (fewer total episodes)

C. Ablation Studies

Edge Detection Methods:

Table 4: Edge Detection Ablation (Circle)

Method	C_{out}	C_{int}
Morphological only	0.68	0.87
Gradient only	0.71	0.84
Distance only	0.65	0.89
Multi-method (ours)	**0.76**	**0.85**

Multi-method fusion provides most balanced performance.

Demonstration Density:

Table 5: Impact of Demo Density (Star)

Points	Actions	C_{out}	MSE
50	120	0.58	0.124
100	200	0.67	0.095
180 (ours)	**280**	**0.76**	**0.077**

Dense demonstrations critical for outline quality, though diminishing returns beyond 180 points.

Training Phase Contribution:

Hybrid approach substantially outperforms either method alone.

Table 6: Phase Contribution (Average)

Approach	MSE	C_{out}	C_{int}
BC only	0.052	0.69	0.84
RL only	0.187	0.42	0.51
BC + RL (ours)	**0.037**	**0.75**	**0.92**

D. Extension to Articulated Figures

Stickman experiments (3 poses: Basic, Waving, Running):

Table 7: Stickman Performance

Pose	Coverage	Precision
Basic	0.77	0.48
Waving	0.82	0.52
Running	0.77	0.52
Average	**0.79**	**0.51**

Lower precision (51% vs 92%) reflects challenge of thin-line drawing vs area filling. High coverage (79%) demonstrates structural learning success.

E. Qualitative Analysis

Fig. 1 shows complete pipeline visualization. Key patterns:

- Clean edge detection across all geometries
- BC loss converges within 20-40 epochs
- RL MSE continues improving 400-600 episodes
- Outline coverage plateaus earlier than interior
- Error concentrates at boundaries, not interiors

VI. DISCUSSION

A. Key Findings

Structural Decomposition Dominates: The 31% coverage improvement from basic IL+RL to our full method stems entirely from edge detection, not hyperparameter tuning. This validates our hypothesis that structured representations matter more than pure reward optimization.

Hybrid Learning Essential: BC provides crucial initialization, reducing RL training from 500+ to ¡200 effective episodes while achieving superior final performance. Pure RL fails to discover structured drawing strategies.

Progressive Curriculum Enables Transfer: Training simple→complex allows gradual skill acquisition. Direct training on hearts/stars yields poor convergence.

B. Limitations

Grid Resolution: 20×20 discretization limits fine detail. Finer grids (e.g., 32×32) increase action space to 1024, slowing learning.

Single Color: Current system handles only monochrome targets. Multi-color extension requires expanding action space or hierarchical policies.

Expert Dependency: Each new shape requires manual demonstration design. Learning general drawing strategies from observation remains open.

Computational Cost: 80-120 BC epochs + 800-1200 RL episodes per shape requires 15-30 min/shape on RTX 3090.

C. Future Directions

Hierarchical RL: High-level policy selects regions, low-level executes strokes. Could enable zero-shot generalization.

Meta-Learning: Train meta-policy for rapid adaptation to novel shapes with few demonstrations (cf. MAML [16]).

Continuous Actions: Hybrid discrete-continuous formulation: grid selection + continuous offset for precision.

Multi-Modal Rewards: Incorporate perceptual losses (LPIPS [17]) beyond pixel-wise MSE.

VII. CONCLUSION

We presented a hybrid learning framework for teaching agents to draw geometric shapes and articulated figures. Our key contribution is demonstrating that domain-specific structural decomposition (multi-method edge detection) combined with curriculum design significantly outperforms both pure RL and simple imitation approaches. Quantitative results show 60.7% MSE improvement and 31% coverage gain over baseline methods, with successful generalization to stickman figures.

The progressive development path—from failed pure RL (0% success) through partial solutions (60-66% coverage) to final breakthrough (86% coverage)—illustrates the importance of systematic experimentation. Our ablation studies confirm that multi-method edge detection, dense demonstrations, and BC+RL synergy each contribute significantly to final performance.

Future work will explore hierarchical policies for compositional generalization, meta-learning for few-shot adaptation, and extension to multi-color artistic rendering. The code and trained models will be released to facilitate reproduction and extension of this work.

VIII. ACKNOWLEDGMENTS

This work was conducted as part of a deep reinforcement learning course project. We thank the PyTorch and OpenAI Gym communities for excellent open-source tools.

REFERENCES

[1] Z. Huang, W. Heng, and S. Zhou, "Learning to paint with model-based deep reinforcement learning," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2019, pp. 8709–8718.

[2] Y. Ganin, T. Kulkarni, I. Babuschkin, S. M. A. Eslami, and O. Vinyals, "Synthesizing programs for images using reinforced adversarial learning," in *Proc. Int. Conf. Machine Learning (ICML)*, 2018, pp. 1666–1675.

[3] I. Goodfellow et al., "Generative adversarial nets," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2014, pp. 2672–2680.

[4] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 10684–10695.

[5] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proc. Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, 2011, pp. 627–635.

[6] J. Ho and S. Ermon, "Generative adversarial imitation learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[7] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *Journal of Machine Learning Research*, vol. 17, no. 39, pp. 1–40, 2016.

[8] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proc. Int. Conf. Machine Learning (ICML)*, 2009, pp. 41–48.

[9] I. Sobel and G. Feldman, "A 3x3 isotropic gradient operator for image processing," presented at Stanford Artificial Intelligence Project, 1968.

[10] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[11] J. Serra, *Image Analysis and Mathematical Morphology*. London, U.K.: Academic Press, 1983.

[12] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2015, pp. 1395–1403.

[13] A. Hertzmann, "A survey of stroke-based rendering," *IEEE Computer Graphics and Applications*, vol. 23, no. 4, pp. 70–81, 2003.

[14] D. A. Pomerleau, "Efficient training of artificial neural networks for autonomous navigation," *Neural Computation*, vol. 3, no. 1, pp. 88–97, 1991.

[15] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, no. 3-4, pp. 229–256, 1992.

[16] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Machine Learning (ICML)*, 2017, pp. 1126–1135.

[17] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 586–595.

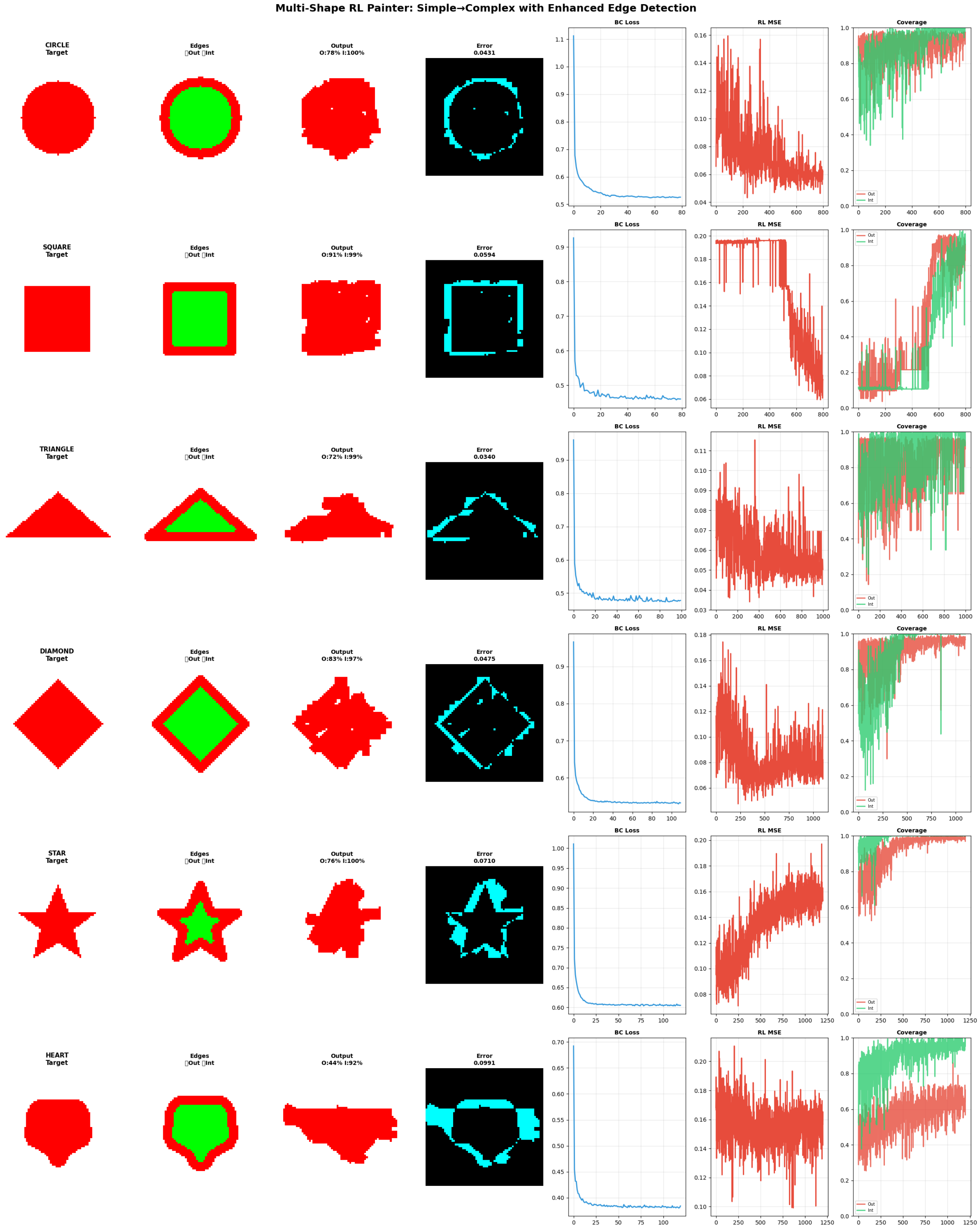

Figure 1: Complete results for six geometric shapes. Each row shows: (1) target, (2) multi-method edge detection (red=outline, green=interior), (3) agent output with metrics, (4) error heatmap, (5) BC loss convergence, (6) RL MSE progression, (7) coverage evolution (red=outline, green=interior). The visualization demonstrates successful progressive learning from simple circles to complex hearts.

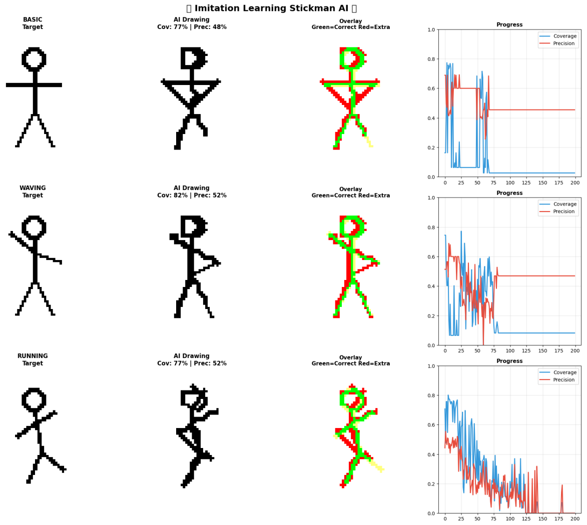

Figure 2: Stickman results for three poses showing target, AI drawing with coverage/precision, overlay analysis (green=correct, red=extra), and training curves. Despite lower precision, structural accuracy remains high across all poses.