Towards GUI Agents: Vision-Language Diffusion Models for GUI Grounding

Supplementary Material

Table 1. Zero Shot performance of LLaDA-V 8B on the Mind2Web test set

Inference Parameters			Accuracy	
Diff. Steps	**Gen Len**	**Block Len**	**SSR (%)**	**F1 (%)**
32	32	32	0.00	0.10
64	64	64	0.00	0.12
128	128	128	0.00	0.10
256	64	64	0.00	0.10

1. Discussion and Limitations

While our work proposes an improved grounding technique for diffusion vision–language models, it remains an early step toward bridging the gap with AR approaches. Although diffusion models demonstrate promising structured grounding capabilities, they still lag behind AR models in latency and accuracy, possibly due to the latter's extensive grounding-specific pretraining and optimized decoding strategies. Our current setup focuses on single-step action prediction, where the output length is short and the latency difference remains modest. However, extending to multi-step action prediction, which involves longer output sequences, will likely present different outcomes. Future work should explore grounding-specific pretraining, efficient diffusion decoding strategies, and architectural refinements to scale diffusion-based models toward multi-action GUI grounding.

2. Appendix B

We evaluated the performance of LLaDA-V with zero-shot prompting. We observed that the model had near-zero performance for the bounding box and action prediction as shown in Table 1

3. Appendix C

We trained LLaDA-V on 7k samples of the Mind2Web train split. For this study, we didn't use OCR-based annotation and cropped images. The sole purpose of this study was to verify if LLaDA-V has potential for GUI grounding and action prediction. We tested the model with different inference parameters and observed that keeping a value of 64 for all inference parameters gives us the best results. The reason is that the model needs a sequence length of 64 as it represents the max length of the target sequence in the test set. The results are shown in Table 2

Table 2. LLaDA-V 8B fine-tuned only on the Mind2Web training set (7k samples) without cropping and OCR based annotation. The table shows the effect of inference parameters on Avg Latency and Accuracy.

Inference Parameters				Accuracy		Avg Lat
Diff. Steps	**Gen Len**	**Block Len**	**Conv Steps**	**SSR (%)**	**F1 (%)**	**sec**
32	32	32	13	78.15	99.00	2.56
64	64	64	25	80.67	99.00	4.84
128	128	128	25	80.63	99.87	5.01
256	64	64	25	80.69	99.87	4.84

4. Appendix D

We evaluate model performance using two metrics: **Action-Type F1** and **Step Success Rate (SSR)**.

Action-Type F1. The Action-Type F1 measures how accurately the model predicts the intended action type among the three possible categories (`lclick`, `hover`, and `type_in`). It is defined in terms of precision and recall as:

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN},$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}},$$

where TP, FP, and FN represent true positives, false positives, and false negatives, respectively. The macro-averaged F1 is computed across all three action types.

Step Success Rate (SSR). The Step Success Rate quantifies how well the predicted bounding box localizes the target element. Let the predicted box be $B_p = (x_1^p, y_1^p, x_2^p, y_2^p)$ and the ground truth box be $B_g = (x_1^g, y_1^g, x_2^g, y_2^g)$. The center of the predicted box is computed as:

$$c_p = \left(\frac{x_1^p + x_2^p}{2}, \frac{y_1^p + y_2^p}{2} \right).$$

A prediction is considered correct if $c_p \in B_g$. The SSR is then defined as:

$$\text{SSR} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\left[c_p^{(i)} \in B_g^{(i)}\right],$$

where $\mathbf{1}[\cdot]$ is the indicator function and N is the total number of evaluated instances.

Table 3. **Accuracy–Latency Trade-Off with Hybrid Masking.** The table compares LLaDA-V trained with default linear masking (third column) and our hybrid masking (fourth and fifth columns) across four benchmarks. The hybrid model achieves higher SSR but with slightly higher latency, while reducing diffusion steps lowers latency with minimal loss in accuracy.

Dataset	Metric	LLaDA-V 8B (Lin)	LLaDA-V 8B (Ours)	
M2W	Conv Steps	16.00	23.00	11.00
	SSR (%)	82.40	83.90	81.00
	F1 (%)	98.50	100.00	96.66
	Lat. (s)	3.02	5.44	2.74
SWI	Conv Steps	18.00	23.00	11.00
	SSR (%)	57.80	63.10	59.60
	F1 (%)	99.50	99.60	98.20
	Lat. (s)	3.36	6.50	2.93
SWT	Conv Steps	17.00	23.00	15.00
	SSR (%)	73.50	74.80	70.00
	F1 (%)	99.10	99.60	90.00
	Lat. (s)	3.20	4.20	3.00
VWA	Conv Steps	16.50	23.00	11.00
	SSR (%)	61.40	67.50	59.20
	F1 (%)	99.40	99.90	99.85
	Lat. (s)	3.05	5.49	2.87

5. Appendix E

We trained LLaDA-V on 7k samples from the Mind2Web training split using OCR-based annotations and cropped images Table 4, following the inference configuration described in Section 3. To study the effect of inference parameters, we varied the number of diffusion steps, generation length, and block length to observe their impact on grounding accuracy and latency. We found that increasing these parameters improves accuracy but also increases latency, with accuracy eventually plateauing or slightly declining when all parameters are raised simultaneously. Interestingly, the converged steps remain lower than the total diffusion steps around 25 on average even when diffusion steps are increased, indicating that LLaDA-V typically reaches confident predictions early in the denoising process.

6. Appendix F

The following subsections present the results obtained using cropped images with OCR-based annotations and the hybrid masking approach, which combines linear and deterministic full masking, to analyze their impact on grounding accuracy and latency.

6.1. Sensitivity to GUI Resolution and Annotation Quality

As shown in Figure 1, we compare predictions from the LLaDA-V model trained with default linear masking, using the Mind2Web train set with and without cropped images and OCR-based annotations. The top example shows the model trained without cropping or OCR annotations. In this case, the model struggles with high-resolution inputs and inconsistent element annotations based on icons of varying sizes, leading to inaccurate grounding of target elements. In contrast, the bottom example shows the model trained with cropped images and OCR-based annotations. Cropping reduces visual complexity, and using text-associated annotations provides more consistent supervision, enabling the model to accurately locate the target element.

6.2. Effect of Hybrid Masking combining Linear and Full Deterministic Masking

As shown in Figure 2, we compare the predictions made by LLaDA-V trained with the default linear masking and our proposed hybrid masking schedule, which combines linear and deterministic full masking(used cropped images and OCR-based annotation for training both variants). The top example shows the model trained with only linear masking, which correctly predicts the action type but produces a slightly misaligned bounding box that does not meet the success criteria. In contrast, the hybrid-masked model (bottom) generates both the correct action and a precisely localized bounding box that satisfies the success condition.

This improvement can be attributed to the structured conditioning introduced by the hybrid schedule. The linear phase enables coarse grounding by predicting the action type and anchor coordinates, while the deterministic phase explicitly conditions the model to refine the bounding-box extent based on the anchor. This targeted supervision helps the model capture geometric dependencies between coordinates, leading to more accurate and spatially consistent predictions across complex GUI layouts.

6.3. Latency Accuracy Tradeoff with Hybrid Masking

Table 3 compares LLaDA-V trained with default linear masking (third column) and our proposed hybrid masking (fourth and fifth columns) across four benchmarks. The hybrid model achieves higher grounding accuracy, with notable SSR gains across all datasets, but also incurs increased latency due to its conditional sequentiality, where the deterministic phase refines predictions based on the linear phase outputs. When diffusion steps are reduced (fifth column), latency decreases substantially with only a slight drop in accuracy, illustrating a clear trade-off between precision and efficiency in diffusion-based grounding.

Table 4. **LLaDA-V 8B inference performance across different inference settings.** The model was fine-tuned on the Mind2Web training set (7k samples) with cropping and OCR-based target annotation, trained for 10 epochs.

Inference Parameters				Accuracy		Latency (seconds)		
Diff. Steps	**Gen Len**	**Block Len**	**Conv Steps**	**SSR (%)**	**F1 (%)**	**Lowest**	**Highest**	**Avg**
8	64	32	4	71.65	99.00	0.99	2.14	1.27
16	64	32	8	74.82	99.00	1.14	2.76	1.47
32	32	32	13	82.50	99.00	2.24	3.43	2.56
64	64	64	25	83.31	99.00	3.72	10.35	4.46
128	128	128	25	82.92	99.00	4.01	28.13	5.02
256	64	64	25	80.61	99.00	3.83	11.03	4.81

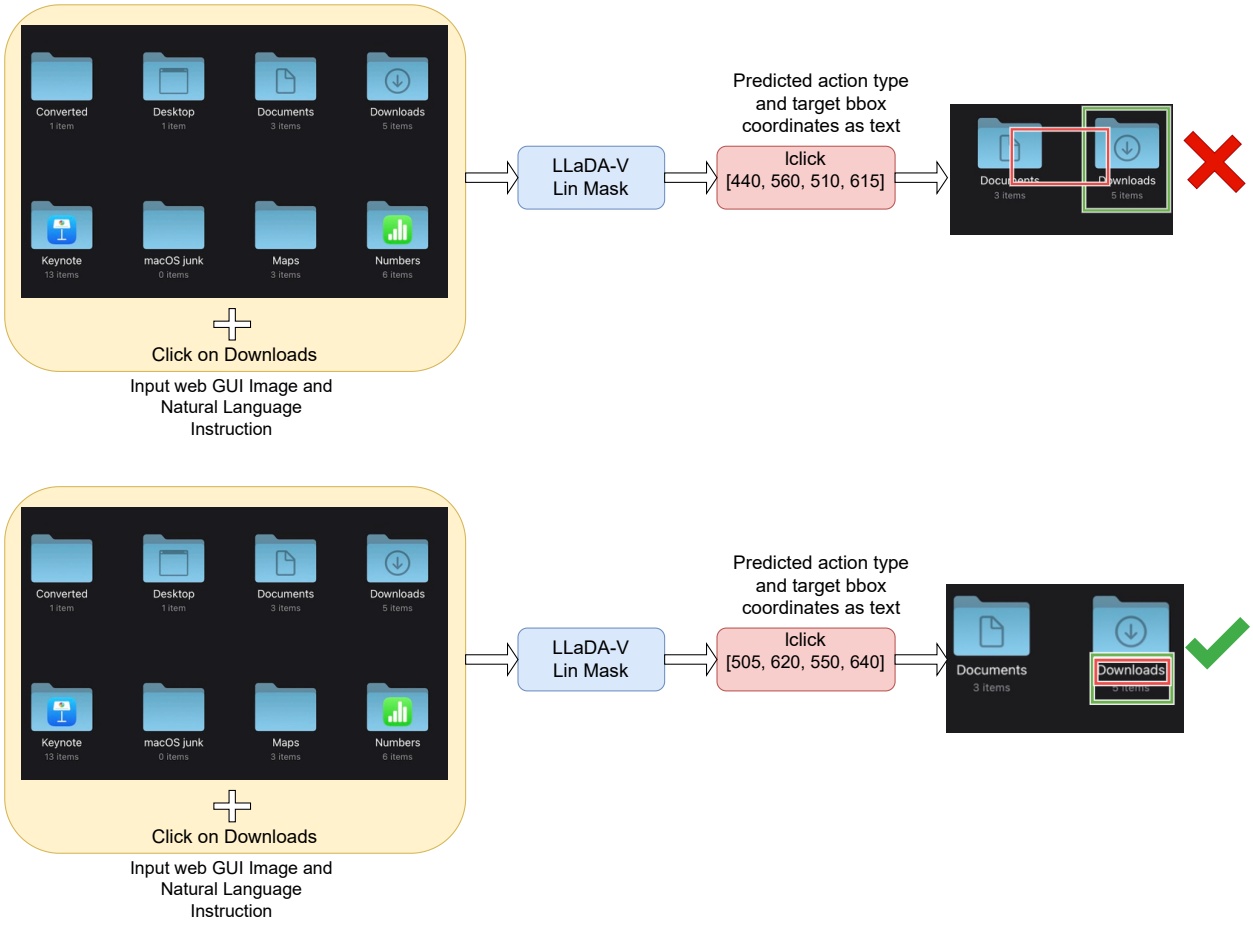

Figure 1. **Effect of data annotation quality on grounding accuracy.** The figure compares predictions from LLaDA-V trained with default linear masking using the Mind2Web train split with and without cropped images and OCR-based annotations. The top example, trained without OCR text-based annotations and cropping, produces an inaccurate bounding box due to inconsistent icon-level targets and high-resolution inputs. The bottom example, trained with cropped images and OCR-guided text annotations, provides more stable supervision, allowing the model to correctly localize the target element. The green bounding box is the ground truth and the red one is the prediction.

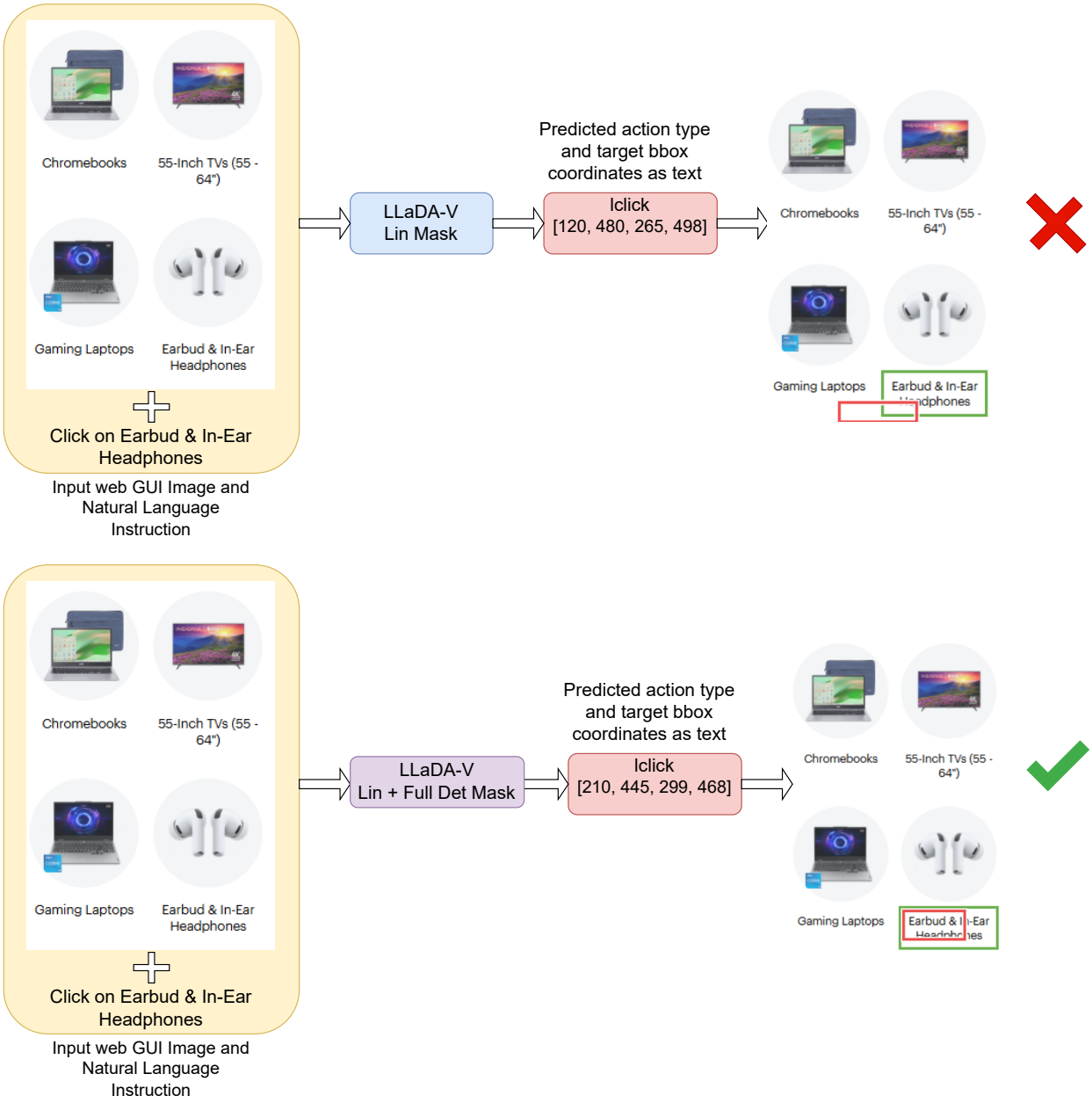

Figure 2. **Effect of hybrid masking on bounding-box accuracy.** The figure compares predictions from LLaDA-V trained with default linear masking (top) and with the proposed hybrid masking that combines linear and deterministic full masking (bottom). The linear-masked model correctly predicts the action type but generates an inaccurate bounding box, missing the target region. In contrast, the hybrid-masked model, guided by conditional refinement between anchor and extent coordinates, produces a precise bounding box that accurately localizes the target element. The green bounding box is the ground truth and the red one is the prediction.