Unifying Layout Generation with a Decoupled Diffusion Model

Mude Hui[1*] Zhizheng Zhang[2] Xiaoyi Zhang[2] Wenxuan Xie[2] Yuwang Wang[3] Yan Lu[2]
[1]Xi'an Jiaotong University [2]Microsoft Research Asia [3]Tsinghua University

Poster ID: TUE-AM-185

Summary

JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

- We present that various layout generation subtasks can be comprehensively unified with a single diffusion model.
- We propose the ***Layout Diffusion Generative Model (LDGM)***, which allows parallel decoupled diffusion processes for different attributes and a joint denoising process for generation with sufficient global message passing and context exploitation. It conforms to the characteristics of layouts and achieves high generation qualities.
- Extensive qualitative and quantitative experiment results demonstrate that our proposed scheme outperforms existing layout generation models in terms of the functionality and performance on different benchmark datasets.

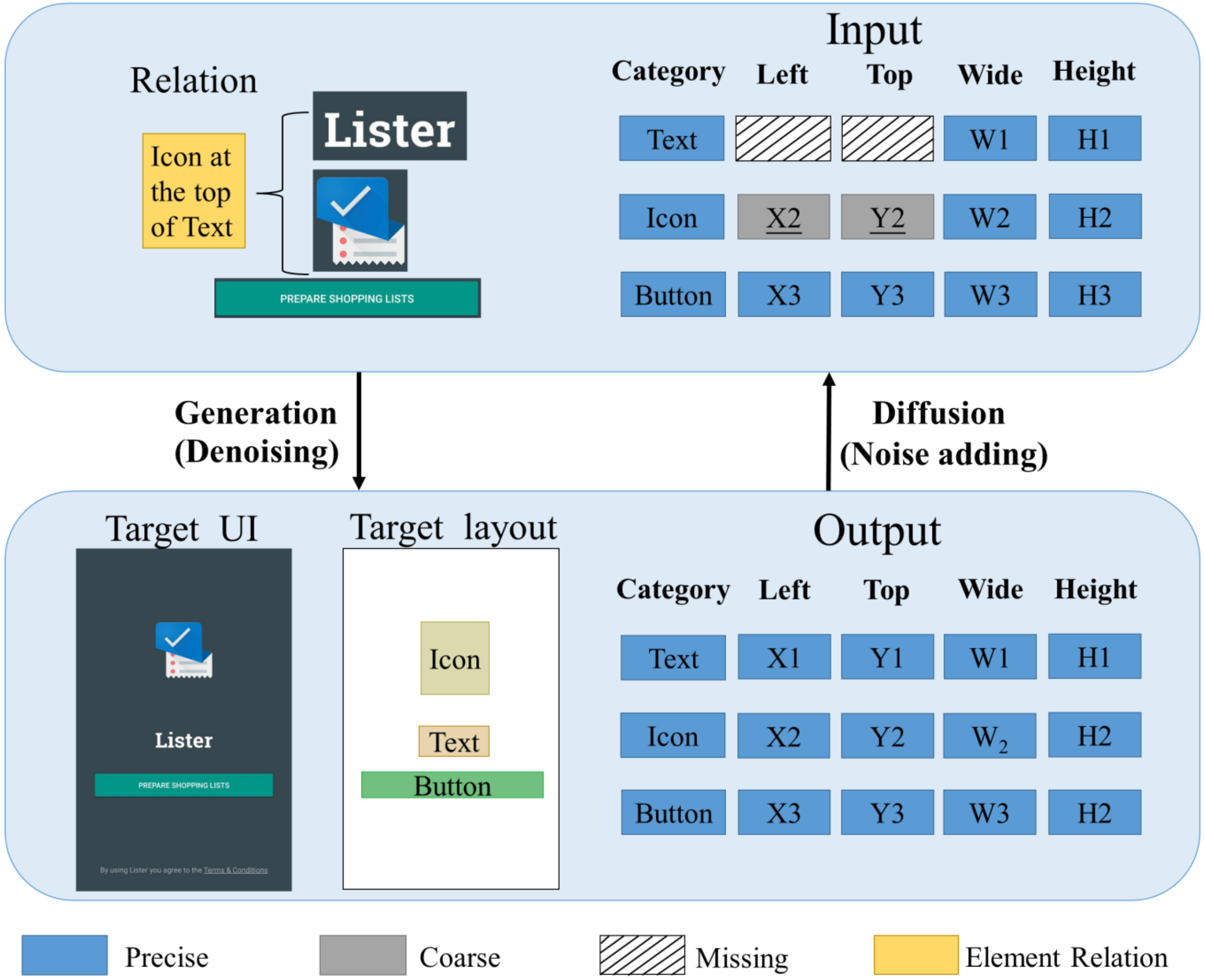
Background

Manual Layout Designs:

- Time-consuming
- Requiring expertise in design

AI-based Layout Generation:

- Diverse demands (versatility)
- Aesthetics & practicality



Generic Settings

U-Gen: unconditional generation

Gen-T: conditioned on types

Gen-TS: conditioned on types & sizes

Gen-TR: conditioned on types & relations

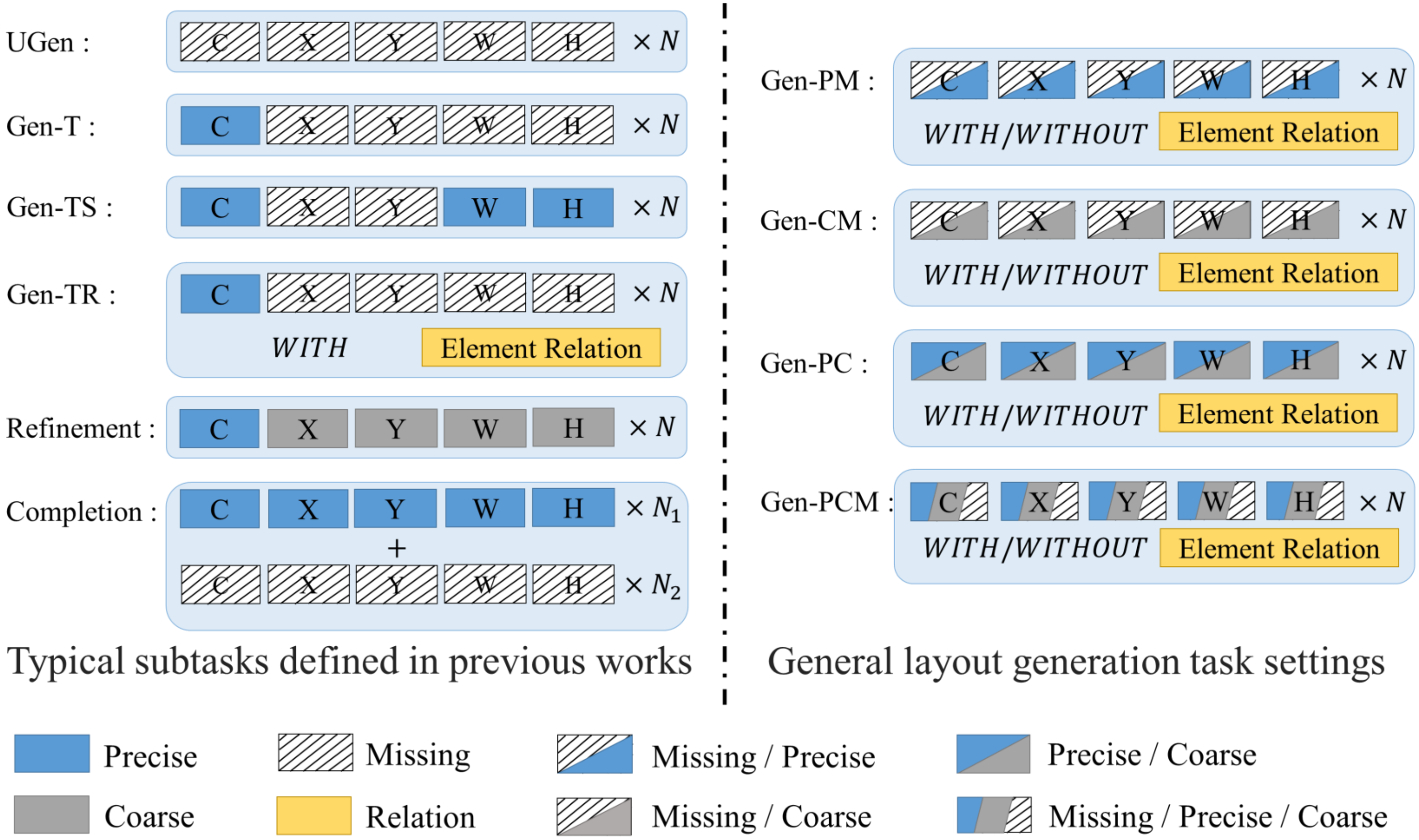
Refinement: update coarse attributes

Completion: generate missing attributes

P: Precise (attributes)

M: Missing (attributes)

C: Coarse (attributes)



Method

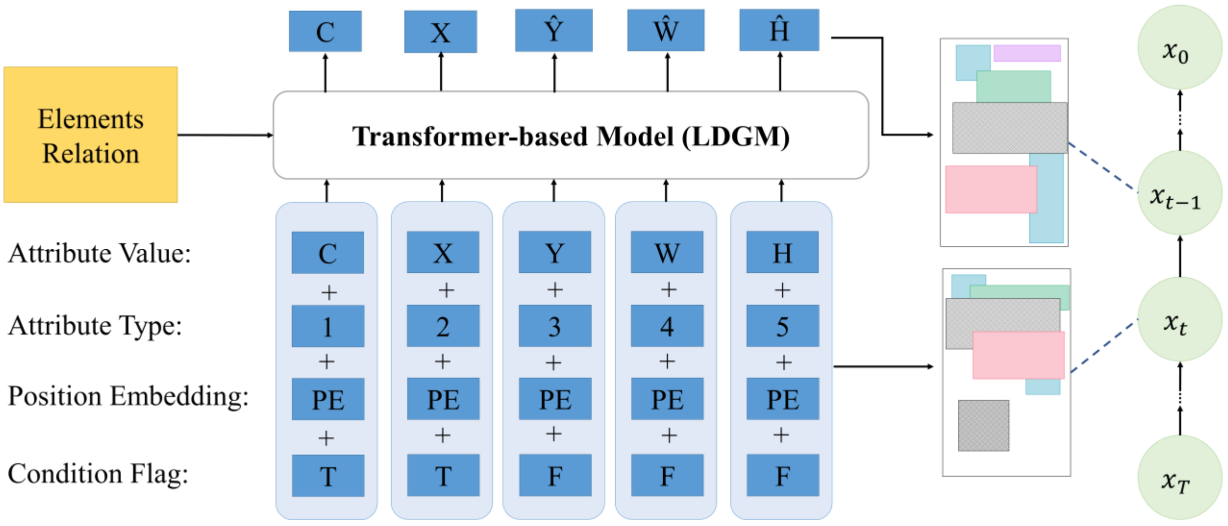
- Unification with Diffusion Modelling

The process from a completed layout to fully corruption. → A diffusion process.

➢ **Layout formulation:**

$$\boldsymbol{l} = [c_1, x_1, y_1, w_1, h_1, c_2, x_2, y_2, \cdots, h_N, \mathcal{E}]$$

➢ **Framework:**



Method

❑Decoupled Diffusion (Training)

Algorithm 1 Training of the LDGM

Require: Transition matrices $\{Q_t^c, Q_t^p, Q_t^s\}$, initial network parameters θ, loss weight λ, and learning rate η.

repeat
- $l \leftarrow$ sample a layout from the training set
- $timsteps = \text{zeros}(\text{len}(l))$ ▷ Record t of attributes.
- $\hat{l} = \text{RandSelect}(l)$ ▷ Select attributes for diffusion.
- $\hat{l} = [C, P, S]$ ▷ Group $\hat{l}$ upon the semantics.
- **for** g in $[C, P, S]$ **do**
  - sample $t \sim \text{Uniform}(\{1, \cdots, T\})$
  - **for** x in g **do**
    - $timsteps[x.index] = t$
    - $x = x_t \leftarrow$ sample from $q(x_t|x_0)$ ▷ Eqn. 2
  - **end for**
- **end for**
- $$\mathcal{L}_x = \begin{cases} \lambda\mathcal{L}_{rec}, & \text{if } timsteps[x.index] = 0 \\ \mathcal{L}_0, & \text{if } timsteps[x.index] = 1 \\ \mathcal{L}_{t-1}, & \text{otherwise} \end{cases}$$
- $\mathcal{L} = \sum_{x \in l} \mathcal{L}_x$
- $\theta \leftarrow \theta - \eta\nabla_\theta\mathcal{L}$ ▷ Update network parameters.

until converged

Different attributes have their own semantics.

Core idea: "***Decouple First Diffusion Then***".

- **For category** c , we adopt noises of a uniform distribution for its diffusion.
- **For position** (x, y) **and size** (w, h), we adopt discretized Gaussian noises for their diffusion.

Method

❑A Joint Denoising Process (Generation Inference)

Algorithm 2 Inference of the LDGM

Require: Initial layout $\boldsymbol{l}_T$, condition flags, and maximum denoising steps T.

$\boldsymbol{l}_T \leftarrow$ tokenize $\boldsymbol{l}_T$ with condition flags
$\boldsymbol{l}_T^m \leftarrow \text{GetMiss}(\boldsymbol{l}_T)$ ▷ Get missing attributes from $\boldsymbol{l}_T$.
$N_m \leftarrow len(\boldsymbol{l}_T^m)$
$k \leftarrow \lceil N_m / T \rceil$
for $t = T, \cdots, 1$ **do**
  $p_\theta(\boldsymbol{l}_{t-1}|\boldsymbol{l}_t) = LDGM(\boldsymbol{l}_t)$
  $\boldsymbol{l}_{t-1}, \boldsymbol{p}_{t-1} \leftarrow$ sample from $p_\theta(\boldsymbol{l}_{t-1}|\boldsymbol{l}_t)$
  if $N_m > 0$ **then**
    $\boldsymbol{l}_{t-1}^m, \boldsymbol{p}_{t-1}^m \leftarrow \text{GetMiss}(\boldsymbol{l}_{t-1}, \boldsymbol{p}_{t-1})$
    $\boldsymbol{l}_{t-1}^m \leftarrow \text{Top-}k\text{Keep}(\boldsymbol{l}_{t-1}^m, \boldsymbol{p}_{t-1}^m)$
    $N_m \leftarrow N_m - k$
  end if
end for
return $\boldsymbol{l}_0$

Handling different attributes of Precise/Missing/Coarse statuses all in one.

- **GetMiss()** refers to an operation of splitting the missing attributes from the entire attribute set.
- **Top-kKeep()** refers to an operation of preserving the predicted results of missing attributes with top-k high confidences and re-mark the remaining ones as absorbing status until all missing attributes are predicted.

Comparison with SOTAs

Subtasks	Methods	Magazine				Rico				PubLayNet			
		MaxIoU ↑	FID ↓	Align. ↓	Overlap ↓	MaxIoU ↑	FID ↓	Align. ↓	Overlap ↓	MaxIoU ↑	FID ↓	Align. ↓	Overlap ↓
U-Gen	LayoutTrans. [7]	0.18	47.84	0.59	47.98	0.46	46.64	0.66	64.10	0.32	49.72	0.37	36.63
	BLT [13]	0.20	44.91	0.55	55.56	0.51	33.81	0.59	67.33	0.34	48.24	0.27	42.79
	UniLayout [9]	0.31	36.61	0.49	**44.50**	**0.62**	26.68	0.40	59.26	0.33	32.29	**0.22**	22.19
	LDGM (Ours)	**0.38**	**32.73**	**0.47**	46.43	**0.62**	**26.06**	**0.36**	**56.35**	**0.46**	**25.94**	0.25	**19.83**
Gen-T	LayoutGAN++ [12]	0.26	36.35	0.54	58.44	0.46	34.43	0.58	59.85	0.36	30.48	0.19	32.80
	BLT [13]	0.22	48.26	0.69	64.01	0.44	39.64	0.57	56.83	0.37	44.86	0.21	38.21
	UniLayout [9]	0.32	28.37	0.51	53.56	0.55	18.06	0.48	57.92	0.41	27.34	0.20	20.98
	LDGM (Ours)	**0.36**	**24.67**	**0.45**	**45.11**	**0.58**	**16.64**	**0.39**	**55.87**	**0.44**	**20.69**	**0.15**	**16.88**
Gen-TS	BLT [13]	0.33	22.72	0.59	61.94	0.51	42.88	0.46	57.74	0.40	24.32	0.16	31.06
	UniLayout [9]	0.35	19.35	0.58	56.43	0.55	20.42	0.49	58.72	0.43	27.47	**0.16**	23.82
	LDGM (Ours)	**0.37**	**17.65**	**0.45**	**44.25**	**0.62**	**12.59**	**0.35**	**55.92**	**0.47**	**19.02**	**0.16**	**10.09**
Gen-TR	CLG-LO [12]	0.27	33.88	0.59	59.43	0.38	38.89	0.54	**56.51**	0.38	31.87	0.21	34.39
	UniLayout [9]	0.36	**19.24**	0.54	49.61	0.57	26.38	0.46	66.93	**0.46**	27.73	0.17	27.35
	LDGM (Ours)	**0.39**	20.58	**0.48**	**47.27**	**0.61**	**16.98**	**0.39**	58.75	0.44	**19.54**	**0.16**	**21.28**
Refinement	RUITE [24]	0.24	44.27	0.64	54.26	0.46	36.70	0.57	64.13	0.32	41.72	0.49	35.74
	UniLayout [9]	0.33	19.78	0.49	49.02	0.56	24.41	0.42	56.04	0.44	22.34	0.11	27.23
	LDGM (Ours)	**0.39**	**14.95**	**0.42**	**37.22**	**0.62**	**13.19**	**0.33**	**52.17**	**0.48**	**15.28**	**0.10**	**13.05**
Completion	LayoutTrans. [7]	0.17	39.36	0.67	55.32	0.46	36.15	0.66	67.10	0.32	41.72	0.37	39.81
	UniLayout [9]	0.23	28.78	0.52	46.43	0.59	25.18	0.45	55.99	0.41	32.04	0.19	22.90
	LDGM (Ours)	**0.38**	**24.35**	**0.49**	**39.26**	**0.60**	**16.42**	**0.36**	**53.15**	**0.44**	**25.31**	**0.10**	**19.45**
Gen-PM	LDGM (Ours)	0.38	27.33	0.47	39.02	0.58	21.64	0.38	56.56	0.46	23.58	0.10	14.11
Gen-CM		0.37	28.74	0.51	43.25	0.57	26.15	0.38	57.74	0.44	24.94	0.11	16.26
Gen-PC		0.37	22.56	0.47	42.95	0.60	18.13	0.36	53.67	0.50	16.42	0.09	12.51
Gen-PCM		0.37	24.45	0.49	44.41	0.59	21.59	0.40	54.77	0.42	25.76	0.14	19.68
GT	-	0.41	9.89	0.43	34.27	0.66	7.05	0.26	49.86	0.64	9.38	0.008	5.18

Generation tasks supported by previous tasks.

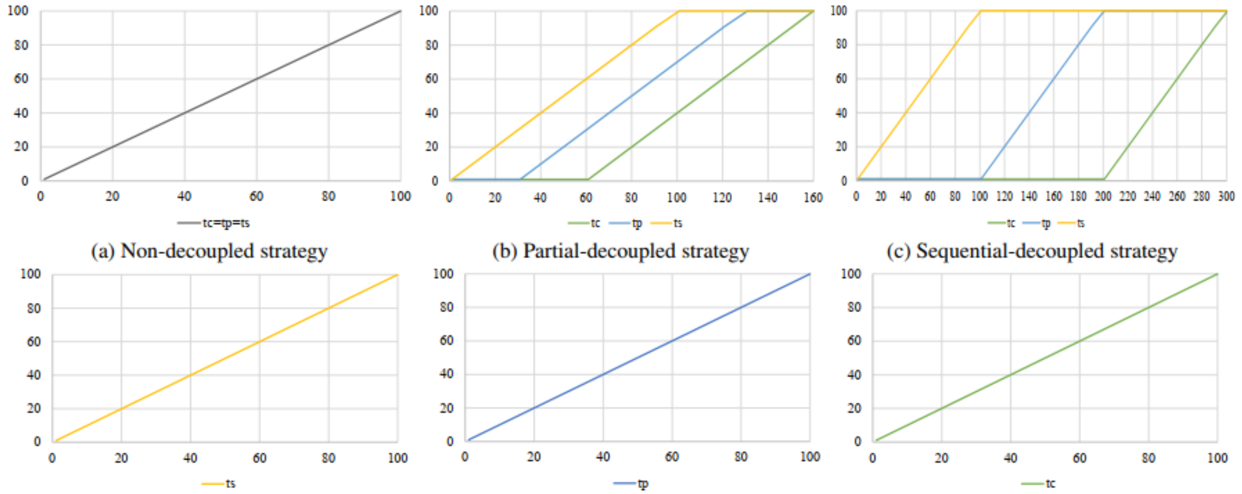
More generic generation tasks supported by ours.

Ablation Studies

JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

- Effectiveness of our proposed decoupled corruption strategy:

Model	MaxIoU ↑	FID ↓	Align. ↓	Overlap ↓
Non-decoupled	0.56	29.24	0.43	60.04
Partial	0.57	27.71	0.48	**54.24**
Sequential	0.56	26.69	0.43	57.17
Parallel (Ours)	**0.59**	**21.59**	**0.40**	54.77

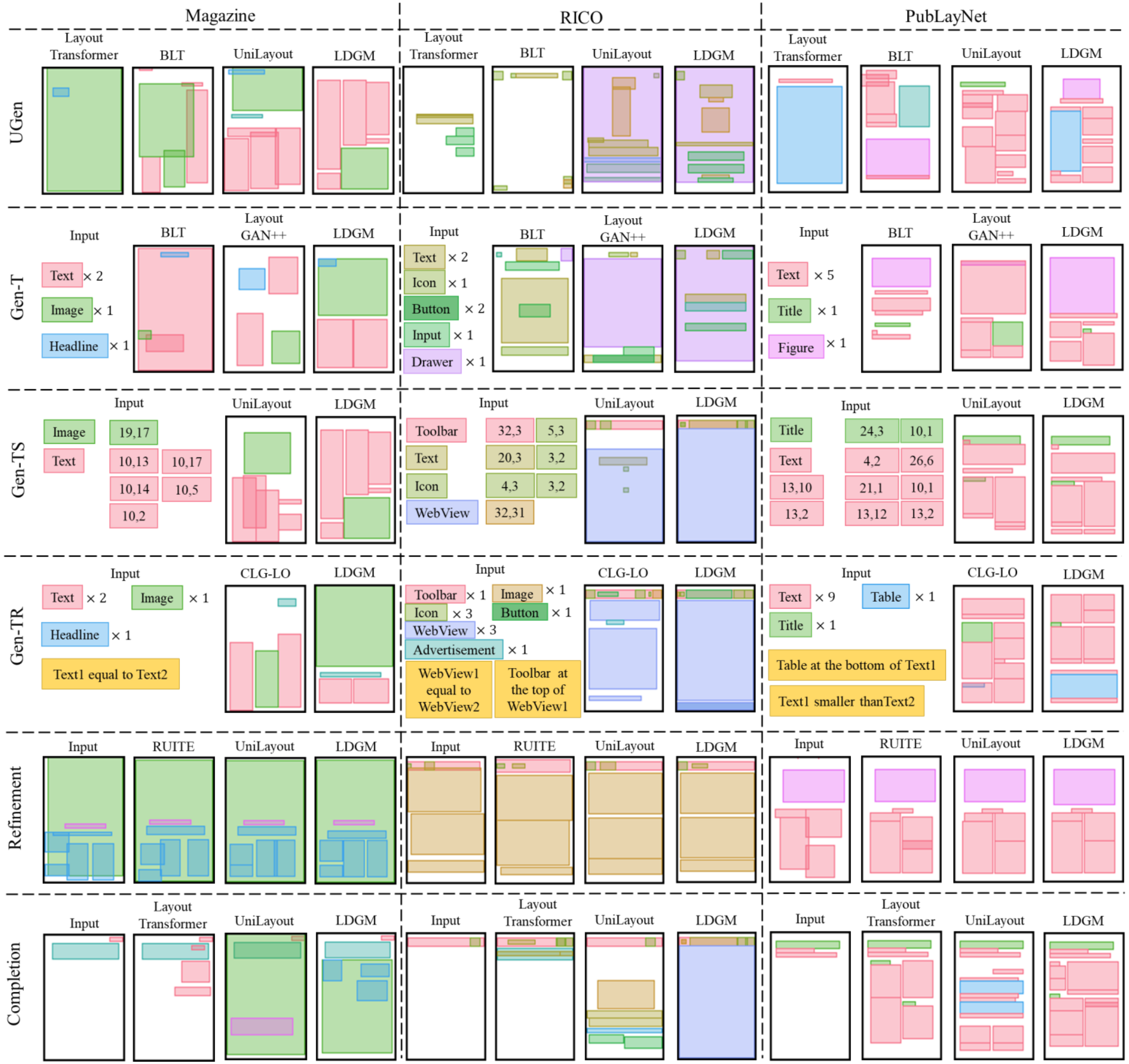


(d) Parallel-decoupled strategy (ours)

- Effectiveness of our proposed inference strategy:

Model	MaxIoU ↑	FID ↓	Align. ↓	Overlap ↓
AutoReg	**0.60**	23.16	0.42	56.87
Non-AutoReg	0.57	25.14	0.44	58.63
Ours	0.59	**21.59**	**0.40**	**54.77**

Visualization



Visualization

JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

Thank You!