Reduced-Complexity Scene Text Recognition Techniques

BUOY RINA

30th June, 2024

Introduction
Text Recognition

- Text – passing down **knowledge**.
- Text recognition – **identifying text** within images.
- Text recognition :
  - Printed text: almost solved (accuracy)
  - **Scene & low-resource non-Latin : challenging** because of diverse imaging conditions and complex text structure.

Scene text images – word level (diverse conditions)

Non-Latin samples – textline level (complex structure and layout)

Introduction

Text Recognition as a Multi-Objective Optimization

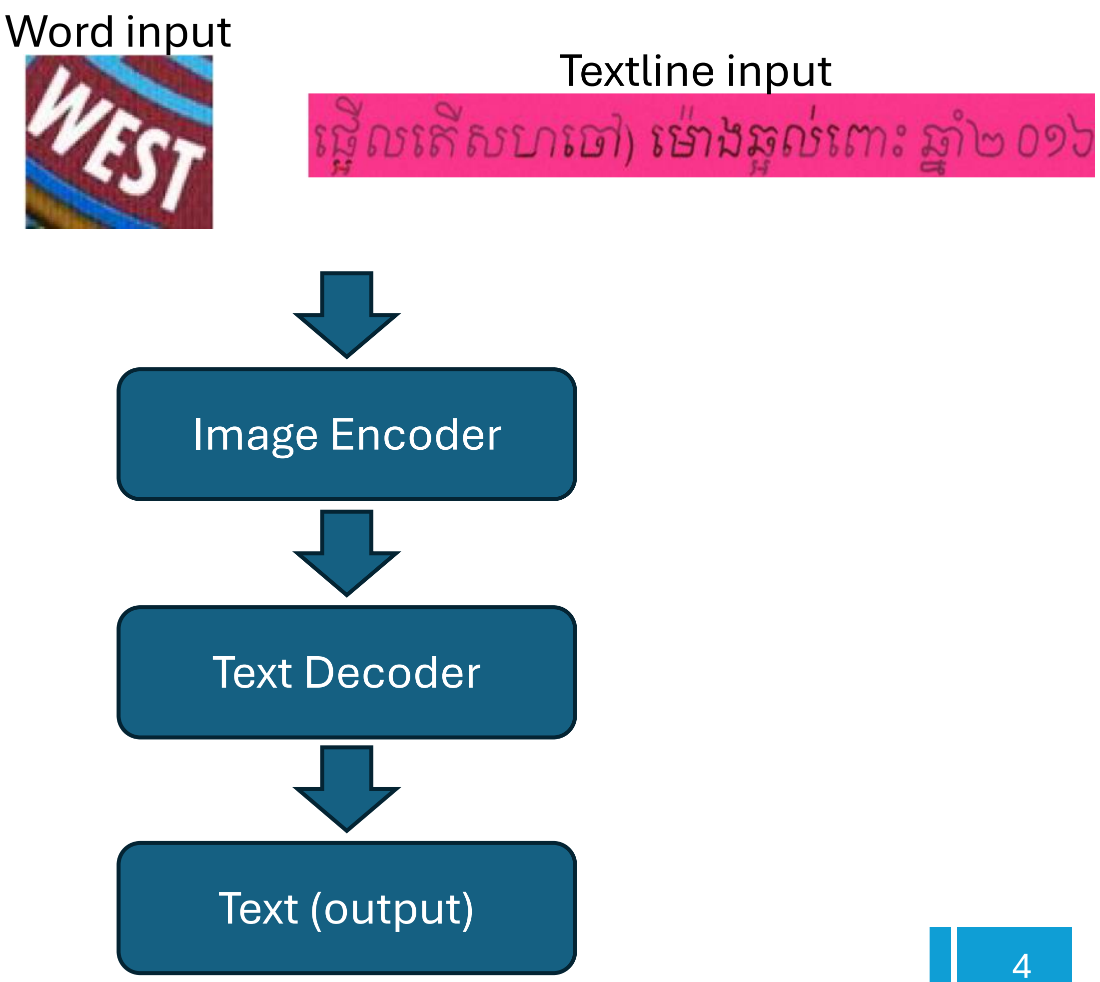
- Many methods – **accuracy-oriented**
- Beyond accuracy↑:
  - **Complexity↓ (latency & memory)** in low-resource settings
  - **Explainability↑ (model understanding)** in safety-critical settings
    - Linking a predicted character to the corresponding image regions
  - **Character location↑**
  - Others
- Optimizing all criteria – **Novel Techniques (this research)**
- **Goal : reducing model complexity↓** and **model latency↓** while **balancing accuracy↑** as well as **enhancing model explainability↑.**

Introduction

Dissecting Text Recognition Methods

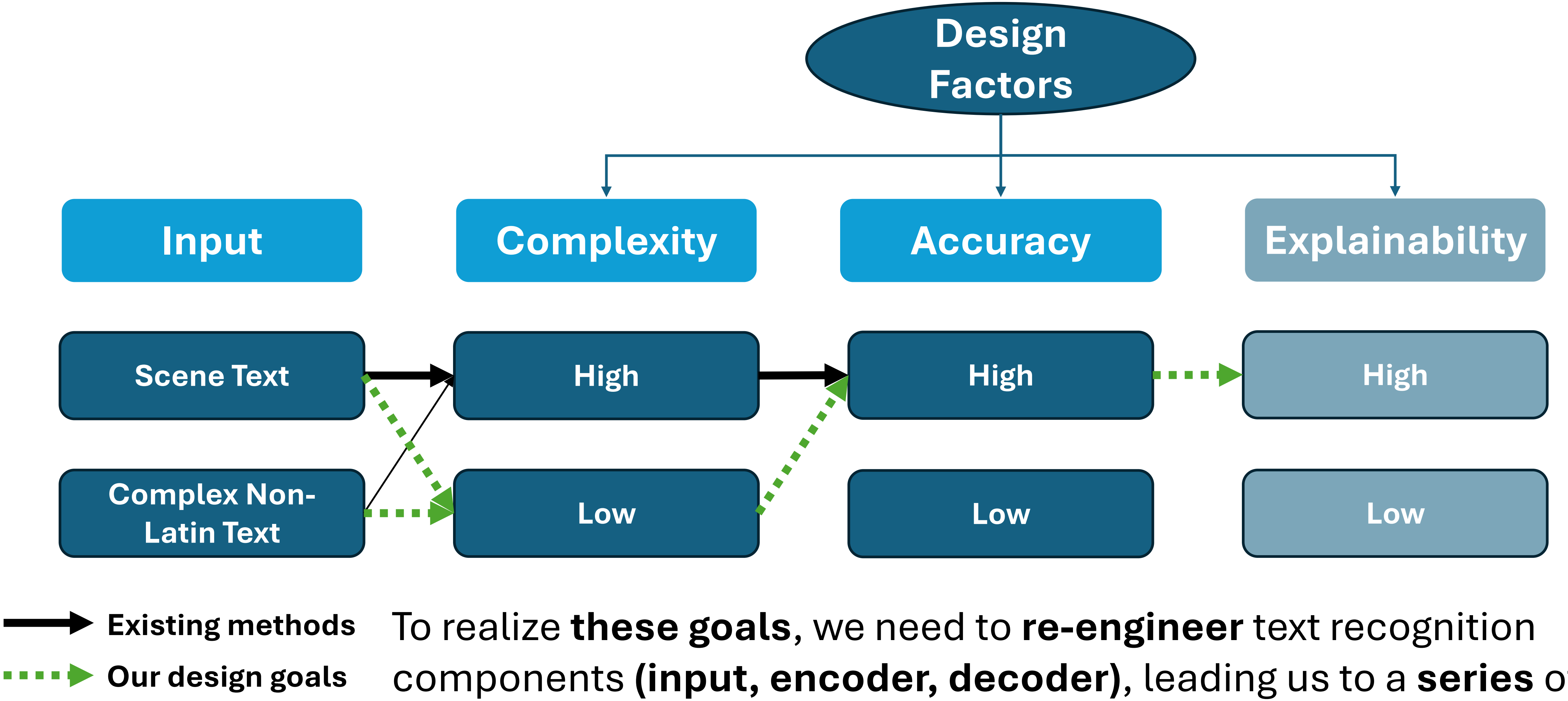
- Text recognition – a **multimodal** problem:
  - Input – **image**
  - Image **encoder** (extracting visual information)
  - Text **decoder** (extracting linguistic information)
  - Output – **text**
- This research aims to **optimize** each **component** to **reduce complexity** and **maximize accuracy**.

Text recognition as a multimodal problem



Introduction

Underlying Design Philosophy

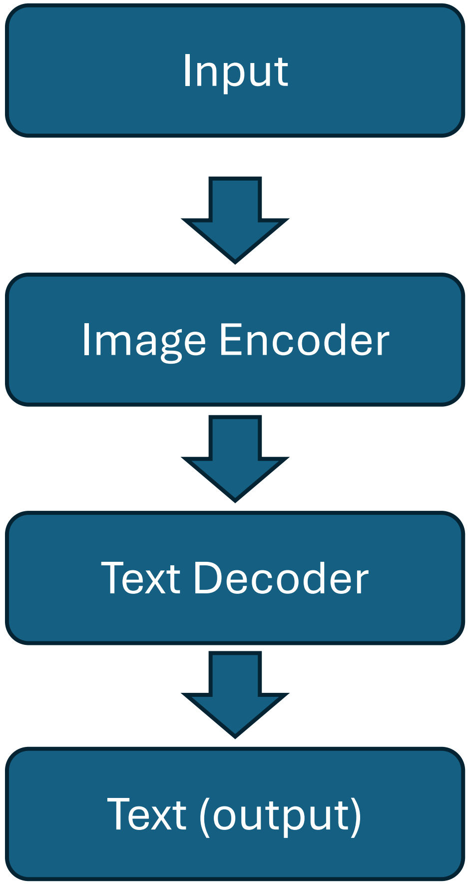


To realize **these goals**, we need to **re-engineer** text recognition components **(input, encoder, decoder)**, leading us to a **series** of **research questions**.

Structure of Presentation

- **Goal**: to reduce **overall model complexity** to make **models efficiently deployed** in **low-resource settings**, while **balancing accuracy and explainability**.
- Achieved by **reducing complexity** & **enhancement** of **each stage**
  - Image encoding – **RQ1**
  - Text decoding – **RQ2**

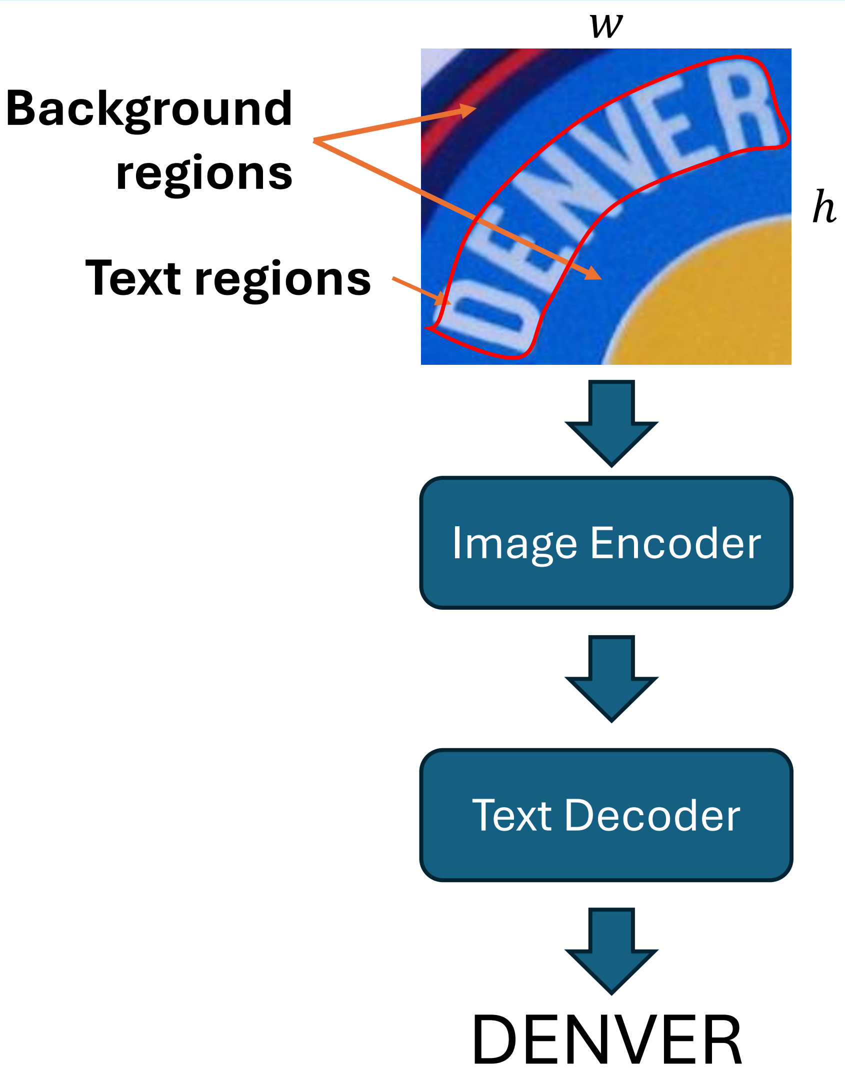
The research questions (RQs) are ordered, following this architecture.



Introduction

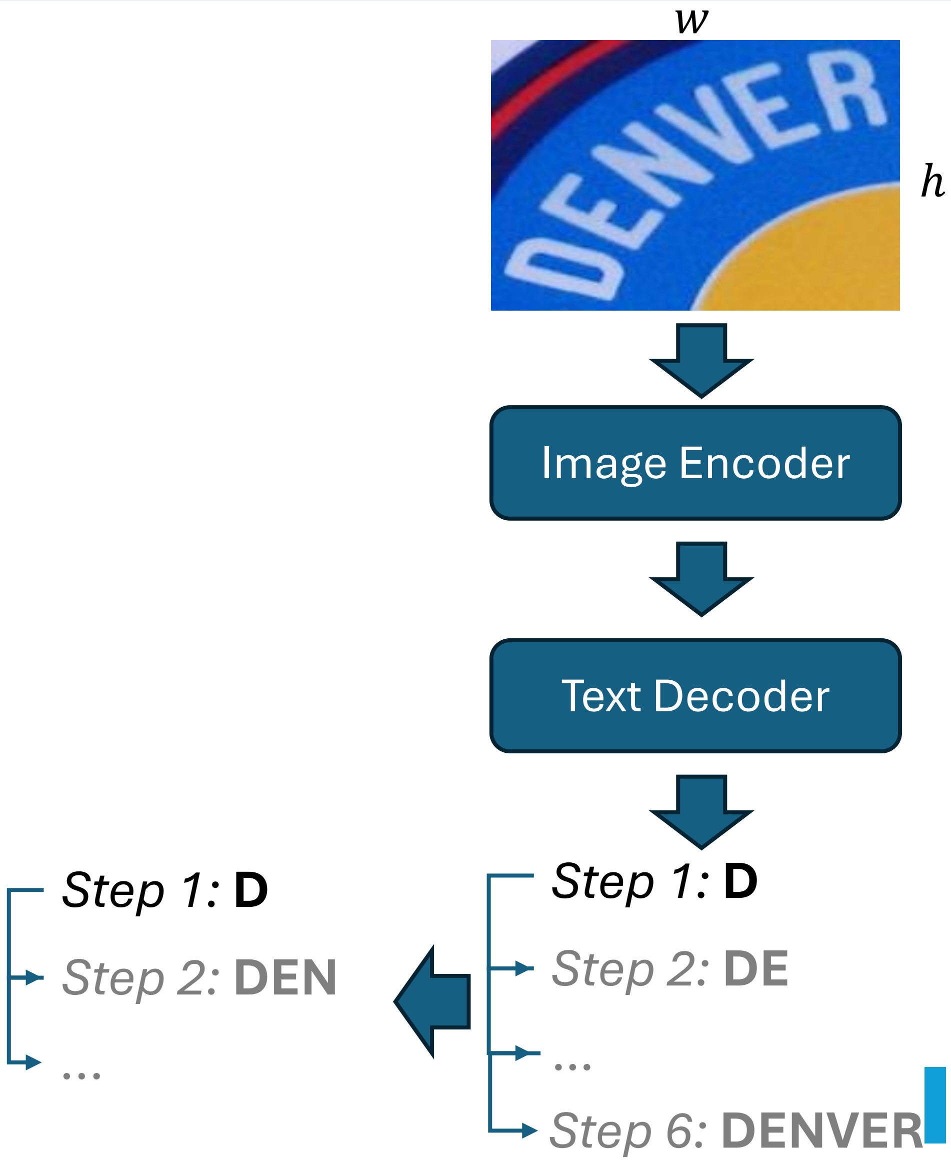
RQ1: ***How to select only text regions?***

- Image encoder:
  - processing **entire input image**
  - **unnecessary complexity** in **background regions**
  - however, only **text regions useful** for **character recognition**
- 1st Research Question (RQ1) : ***How to select only text regions for recognition, and thus, remove unnecessary complexity while balancing accuracy?***



Introduction
RQ2: *How to predict many characters ahead ?*

- Text decoder:
  - **one character at a time** (i.e., **autoregressive** like GPT)
  - but, **high latency** (many decoding steps)
- 2nd Research Question (RQ2) : ***Can we reduce decoding steps by predicting many characters ahead while balancing accuracy?***
  - E.g., predicting ***two or three characters*** together after **D** in a single step.

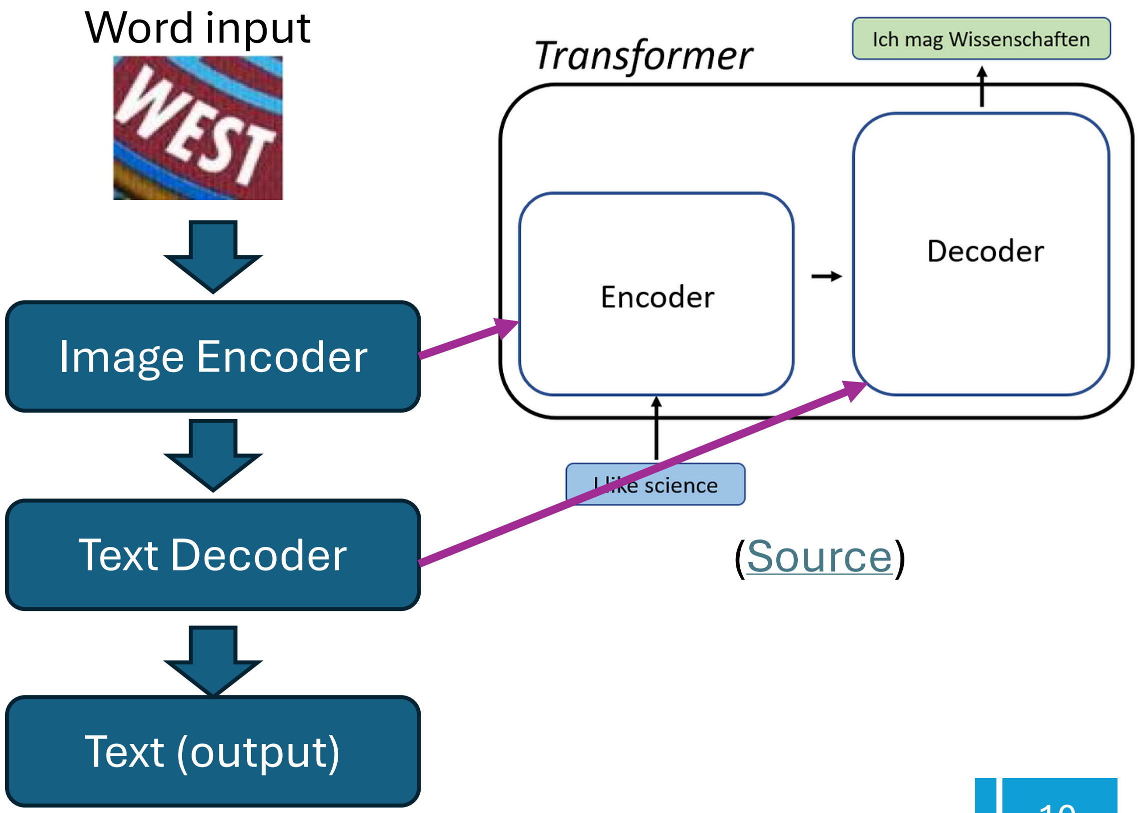


Related Work

Related Work
Latin Scene Text Recognition (STR)

- Well-studied – **Transformers everything, everywhere**
- **Complexity bottlenecks**:
  - Encoder: **quadratic scaling** with **input width**
  - Decoder: **high-complexity cross-attention** & **high-latency** decoding
  - Challenging in a **low-resource** setting, or **long inputs**
- **Bottlenecks** => **motivations** for **our research questions.**

Similarity between text recognition & Transformers architectures.



(Source)

How to Select Only Text Regions?

1^{st} Research Question

Journal Paper:

Towards reduced-complexity scene text recognition (RCSTR) through a novel salient feature selection

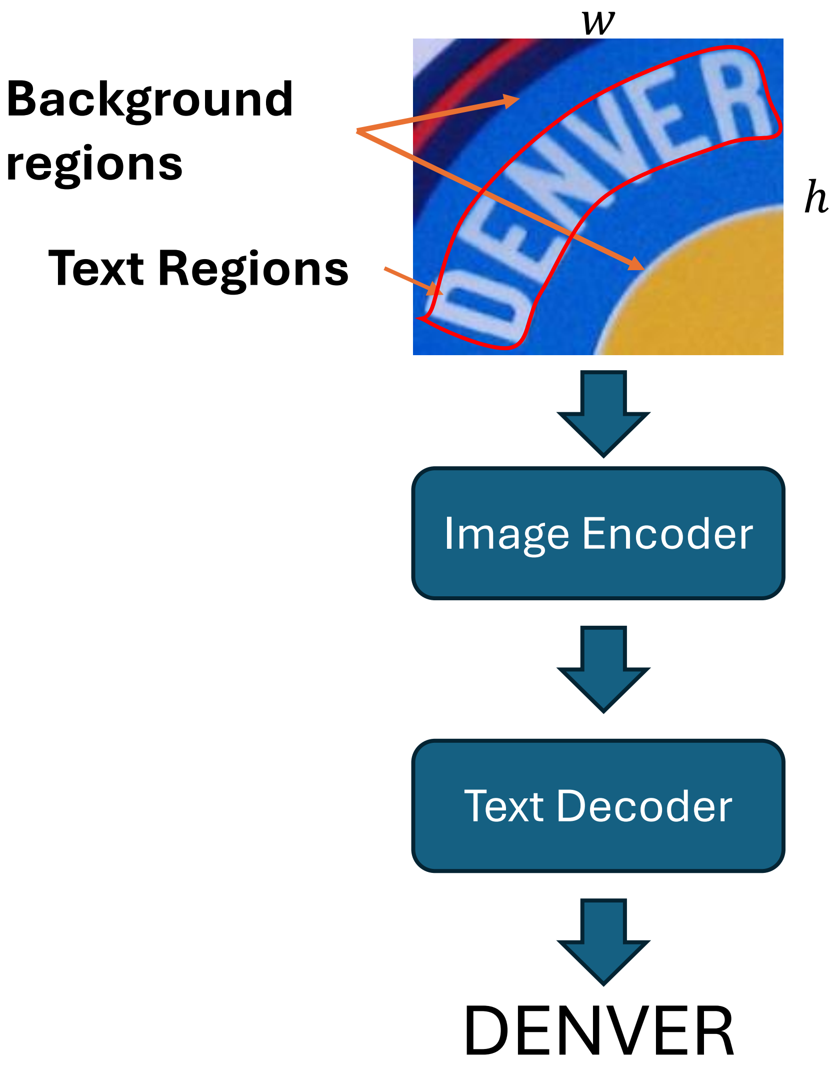
Special Issue Paper | Published: 22 May 2024

(2024) Cite this article

International Journal on Document Analysis and Recognition (IJDAR) & International Conference on Document Analysis and Recognition (ICDAR 2024; Greece)

RQ1: Reduced-Complexity Scene Text Recognition
Introduction

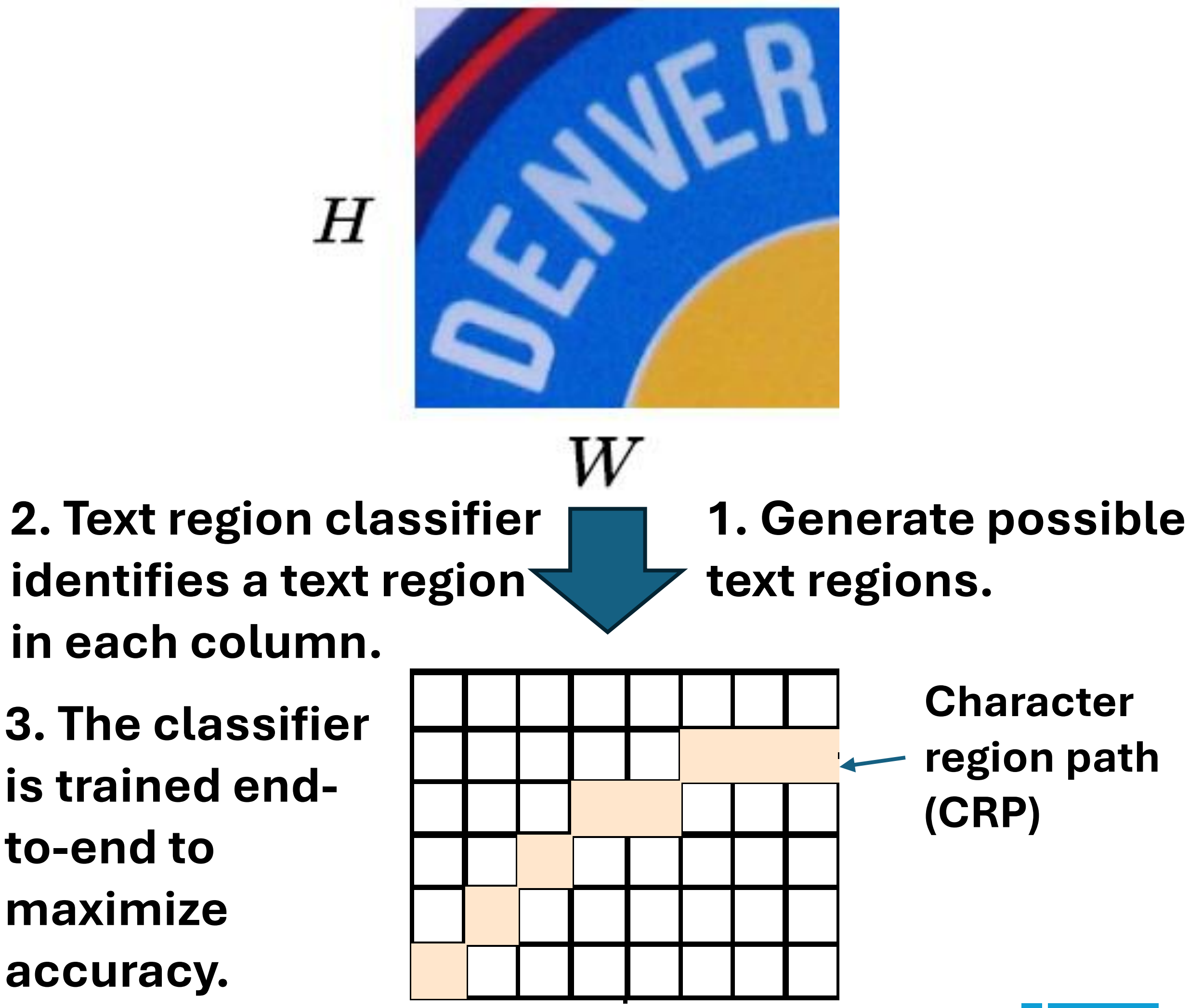
- **Highly-oriented texts**:
  - Small **text regions**.
  - Large **background regions**.
- Processing **entire inputs**:
  - **Extra complexity** in the background region.
- Leading to **high-complexity** for **downstream stages**.
- Is it necessary to process to the **entire inputs** ? Can we use only the **text regions**?



RQ1: Reduced-Complexity Scene Text Recognition

Proposed Method

- **Solution**: a **character region path (CRP)** in the **text regions**.
- **Text regions**:
  - Collected along the **CRP**.
  - Used for **downstream stages** (encoder, decoder).
- **Removing unnecessary computations** in the background regions.
- Thus, **reducing model complexity** as well as **enhancing latency**.



RQ1: Reduced-Complexity Scene Text Recognition
Proposed Method

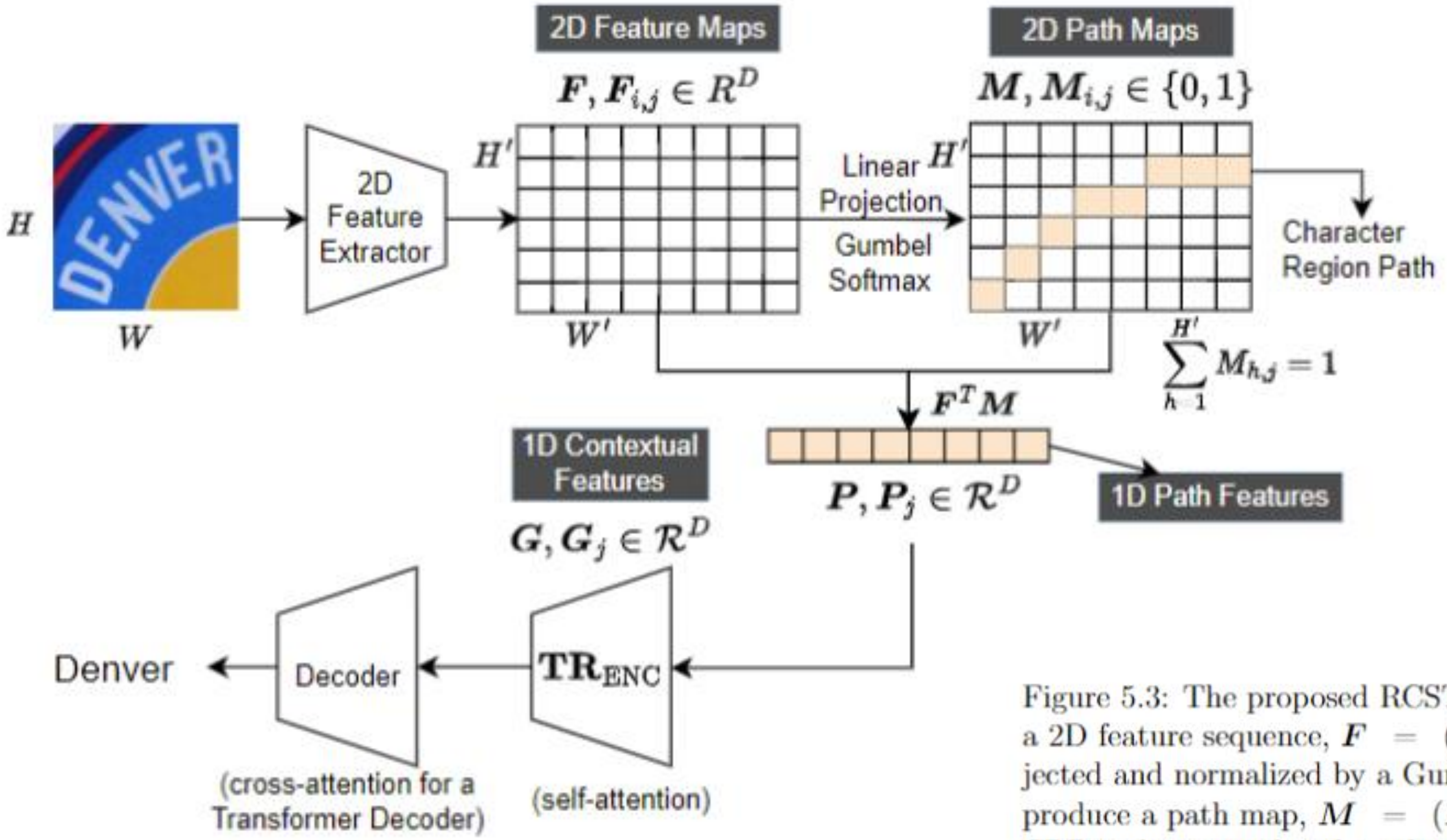


Figure 5.3: The proposed RCSTR framework. A 2D feature extractor produces a 2D feature sequence, $\boldsymbol{F} = (\boldsymbol{F}_{1,1}, ..., \boldsymbol{F}_{H',W'}), \boldsymbol{F}_{i,j} \in \mathcal{R}^D$. $\boldsymbol{F}$ is linearly projected and normalized by a Gumbel-Softmax layer along the H' dimension to produce a path map, $\boldsymbol{M} = (M_{1,1}, ..., M_{H',W'}), M_{i,j} \in \{0,1\}$. Based on $\boldsymbol{M}$, a CRP is determined and a salient subset of the features, $\boldsymbol{P} = (\boldsymbol{P}_1, ..., \boldsymbol{P}_{W'})$, is selected for subsequent processes. D is the model embedding dimension. H and W are the input height and width, respectively. H' and W' are the height and width of feature maps, respectively. Best viewed in color.

RQ1: Reduced-Complexity Scene Text Recognition
Datasets & Training

Training on Synthetic Data

- MJSynth[12,]
- SynthText[13]
- SynthAdd[14]
- SynthTiger[65]

Finetuning on Real Labeled Data

- COCO-Text[96]
- RCTW[97]
- Uber-Text[98]
- ArT[99,]
- LSVT[100]
- ReCTS[101]
- TextOCR[65]
- OpenImages V5[60]

Evaluation

- SVT[89]
- IIIT[90]
- ICDAR2013[91]
- ICDAR2015[92]
- SVTP[93]
- CUTE80[94]

RQ1: Reduced-Complexity Scene Text Recognition
Experimental setup & efficiency comparison

- Encoder:
  - **Convolutional Transformer** as image encoder
- Decoder:
  - **Transformer decoder** as text decoder
- Inputs:
  - using **text regions** only based on the **proposed technique**.
- Baseline model:
  - using **entire inputs**.

FLOPs↓ and inference time↓ comparisons

Model	Params		FLOPs		Time (ms)
	Enc.	Dec.	Enc.	Dec.	
Baseline-TrDec (No CRP)	22.5	12.7	6.1	14.5	270
RCSTR-TrDec (CRP - Ours)	22.5	12.7	3.8	5.4	182

- <u>**Efficiency analysis:**</u>
  - **reducing model complexity** in both encoder (6.1 -> 3.8B FLOPs) and decoder (14.5 -> 5.4B FLOPs).
  - **reducing inference time** by **half (270 -> 182** *ms***)**.

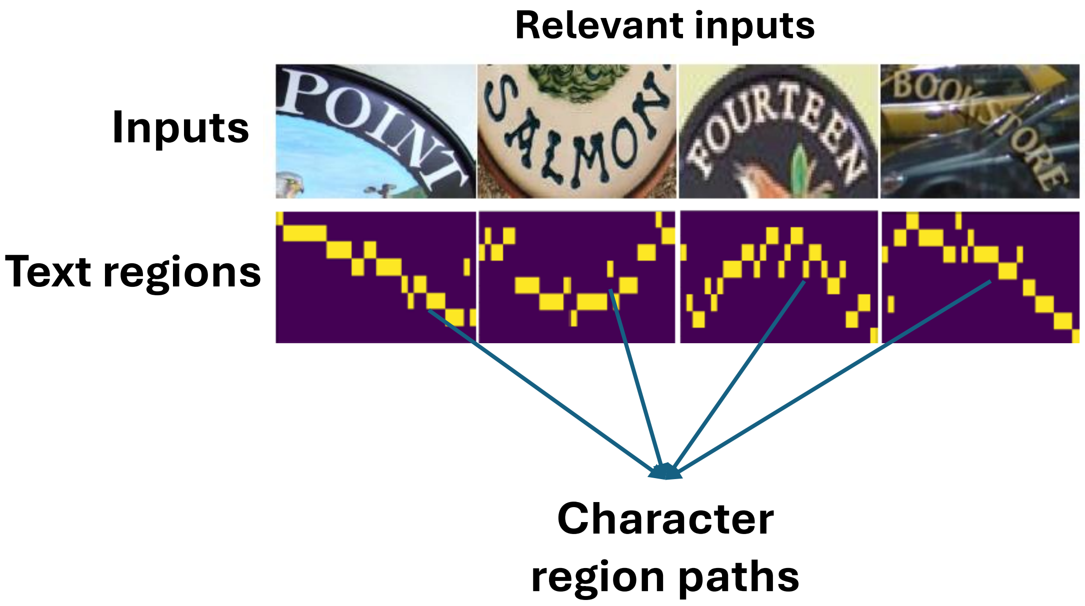
RQ1: Reduced-Complexity Scene Text Recognition
Accuracy comparisons with the baseline model

- Comparing with **the baseline:**
  - **Comparable accuracy** – <0.4%
  - **Two times faster**

Accuracy↑ (%) comparison with the baseline

Method	IIIT	SVT	IC13	IC15	SVTP	CUTE	Total
Baseline-TrDec (No CRP)	97.8	95.7	97.3	89.8	90.5	96.5	94.9
RCSTR-TrDec (CRP - Ours)	97.8	94.9	97.0	89.1	90.2	94.8	94.5

- **Identified text regions:**
  - **Consistent text regions** even for **highly-oriented texts**, thus validating our proposed text regions technique.



RQ1: Reduced-Complexity Scene Text Recognition
Accuracy comparisons with the SOTA methods

- Comparing with the **most recent SOTA method** using entire inputs:
  - **Consistently improving accuracy** or **comparable** (<0.4%).
- Thus, the **proposed technique** can effectively **reduce complexity** by focusing only on the **text regions**.

Accuracy ↑ (%) comparison with the recent existing methods

Method	IIIT	SVT	IC13	IC15	SVTP	CUTE	Total
TRBA [95]	94.8	91.3	94.0	80.6	82.7	88.1	89.6
DiG-ViT-T [23]	96.4	94.4	96.2	87.4	90.2	94.1	93.4
DiG-ViT-S [23]	97.7	96.1	97.3	88.6	91.6	96.2	94.7
DiG-ViT-B [23]	97.6	**96.5**	**97.6**	88.9	**92.9**	**96.5**	**94.9**
RCSTR-TrDec (CRP - Ours)	**97.8**	94.9	97.0	**89.1**	90.2	94.8	94.5

RQ1: Reduced-Complexity Scene Text Recognition
Known limitations

- Character region paths:
  - Text regions collected along the width => not applicable to vertical text.

A vertical text

Can We Predict Many Characters Ahead?

2nd Research Question

Journal Paper:

Parstr: partially autoregressive scene text recognition

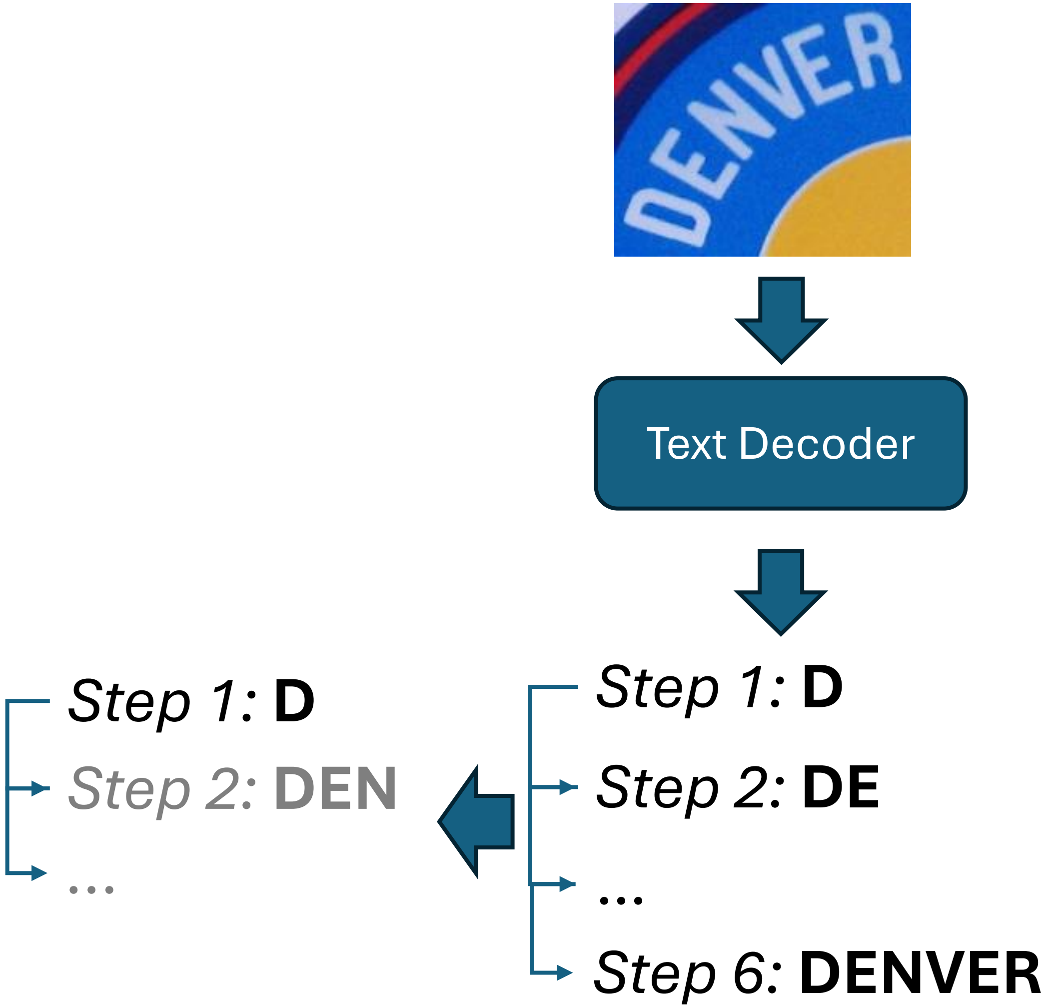
Special Issue Paper | Published: 22 May 2024

(2024) Cite this article

International Journal on Document Analysis and Recognition (IJDAR) & International Conference on Document Analysis and Recognition (ICDAR 2024; Greece)

RQ2: Partially Autoregressive Decoder for STR
Introduction

- Text decoder:
  - **one character at a time** (i.e., **autoregressive** like GPT)
  - but, **high latency** (many decoding steps)
- Instead of predicting **one character** at a time, can we predict **more than one**?
  - E.g., predicting **two or three characters** after ***D***.
- Leading to **reduced-complexity decoding** and, thus **lower latency**.



RQ2: Partially Autoregressive Decoder for STR

Proposed Method

- **Solution** : **innovative decoding strategies** to predict **many characters** in a **single step.**
- Two **proposed decoding** schemes :
  - **b-first**: first b **characters** one at a time, the rest together.
    - **0-first = all at once, n-first = one by one.**
  - **b-ahead**: b **characters** in one step.
    - **0-ahead = one by one, n-ahead = all at once.**

7 characters

3-first ($b = 3$):

- ***COL*** (one at a time – 3 steps)
- ***LEGE*** (one step)

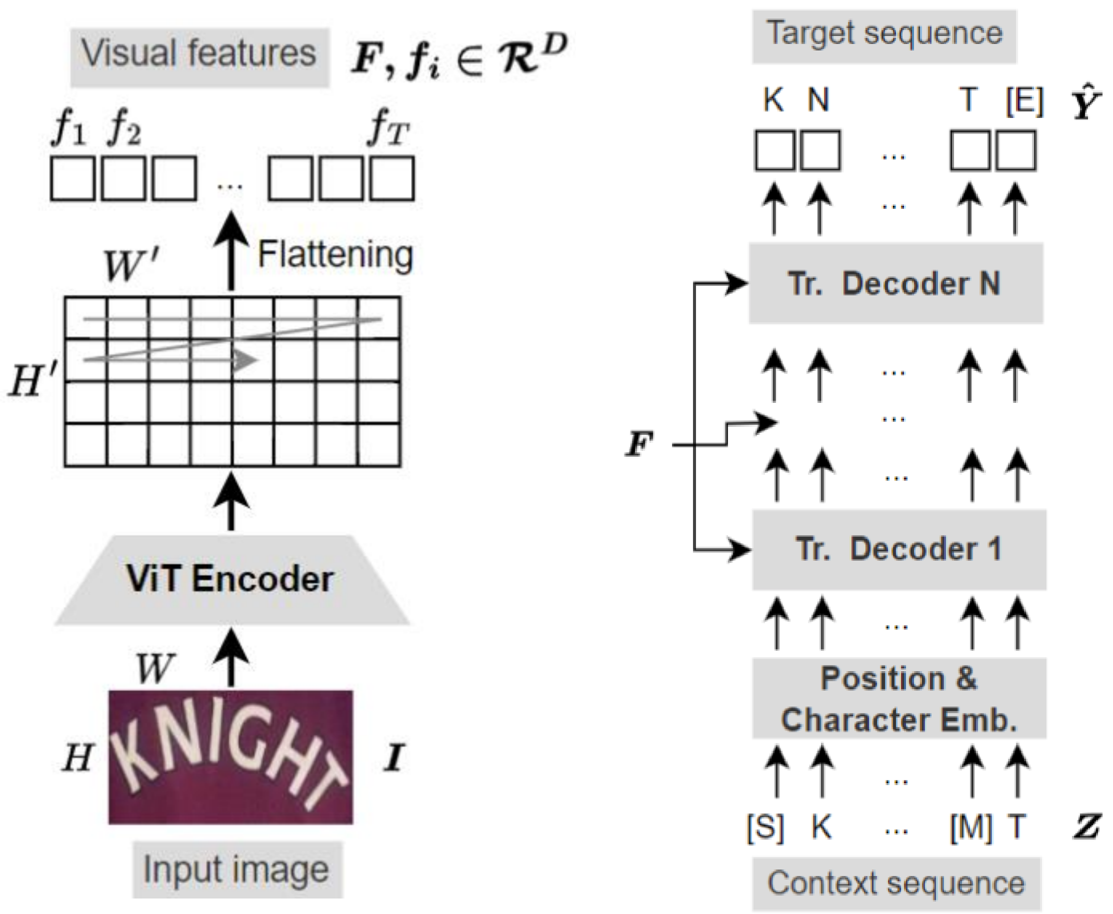
3-ahead ($b = 3$):

- ***COL*** (one step)
- ***LEG*** (one step)
- ***G*** (one step)

RQ2: Partially Autoregressive Decoder for STR
Proposed Method & Model Setup

- Decoding efficiency :
  - $\boldsymbol{b}$**-first:** approximately $\boldsymbol{b}$ **decoding steps**.
  - $\boldsymbol{b}$**-ahead:** approximately **a factor of** $\boldsymbol{b}$
- Proposed decoder: **partially autoregressive decoder (PAR)**
- **Model Setup:**
  - Encoder: vision Transformer as image encoder
  - Decoder: our proposed **PAR decoder** as text decoder

RQ2: Partially Autoregressive Decoder for STR
Proposed Method & Model Setup



	[B]	K	N	I	G	H	T
K	1	0	0	0	0	0	0
N	1	1	0	0	0	0	0
I	1	1	1	0	0	0	0
G	1	1	1	1	0	0	0
H	1	1	1	1	1	0	0
T	1	1	1	1	1	1	0
[E]	1	1	1	1	1	1	1

(a) AR left-to-right decoding

	[B]	[M]	[M]	[M]	[M]	[M]	[M]
K, N, I, G, H, T, [E]	1	1	1	1	1	1	1

(b) NAR parallel decoding

RQ2: Partially Autoregressive Decoder for STR

Proposed Method & Model Setup

$$
\begin{array}{r|ccccccc}
 & [B] & K & N & [M] & [M] & [M] & [M] \\
\hline
K & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
N & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
I, G, H, T, [E] & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{array}
$$

(a) b-first decoding

$$
\begin{array}{r|ccccccc}
 & [B] & [M] & [M] & [M] & [M] & [M] & [M] \\
\hline
K, N, I & 1 & 1 & 1 & 0 & 0 & 0 & 0
\end{array}
$$

$$\downarrow$$

$$
\begin{array}{r|ccccccc}
 & [B] & K & N & I & [M] & [M] & [M] \\
\hline
G, H, T & 1 & 1 & 1 & 1 & 1 & 1 & 0
\end{array}
$$

$$\downarrow$$

$$
\begin{array}{r|ccccccc}
 & [B] & K & N & I & G & H & T \\
\hline
[E] & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{array}
$$

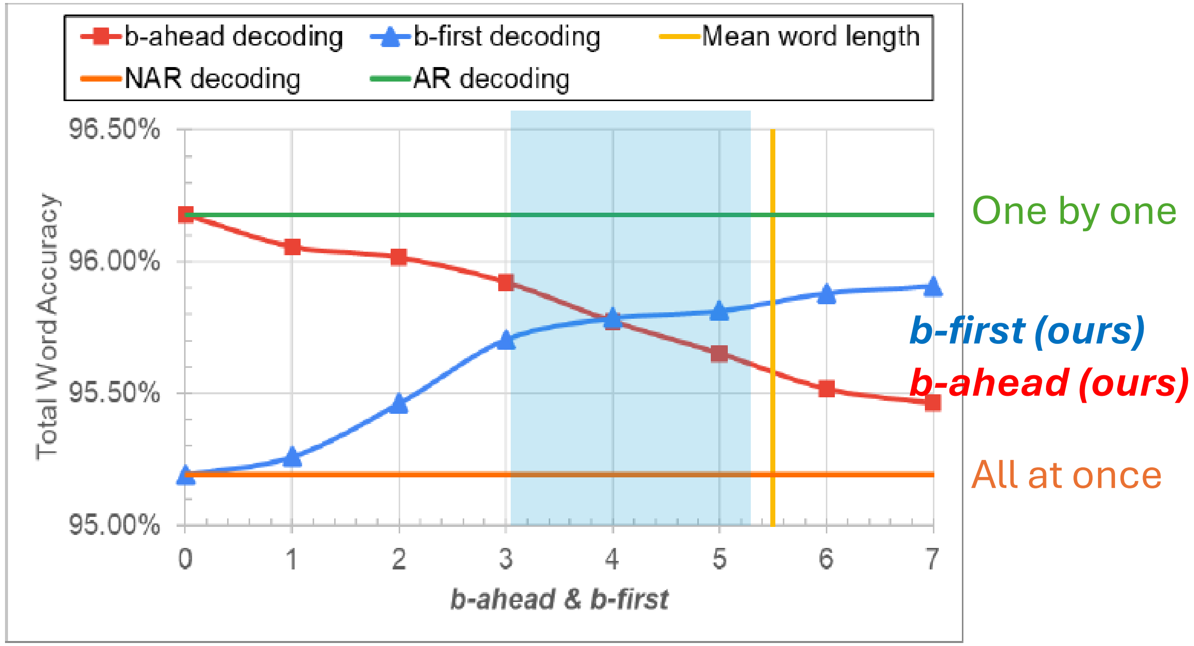
(b) b-ahead decoding

Figure 7.6: The b-first vs. b-ahead decoding steps for the target text, *KNIGHT*. (a) b-first with $b = 2$. (b) b-ahead with $b = 3$: in each generation, three characters are decoded and used in the subsequent steps. The first row is the context sequence and the first column is the target sequence. 1 indicates allowing attention, while 0 indicates the opposite. The blue mask values correspond to the blue characters, which are decoded. Best viewed in color.

RQ2: Partially Autoregressive Decoder for STR
Experimental setup & efficiency analysis

- **Accuracy analysis:**
  - between all-at-one decoding (one step) and one-by-one decoding (all steps)
  - ***b*-first** : b increases, accuracy increases.
  - ***b*-ahead**: b increases, accuracy decreases.
  - **Optimal range – 3 to 5 (<0.5%)**

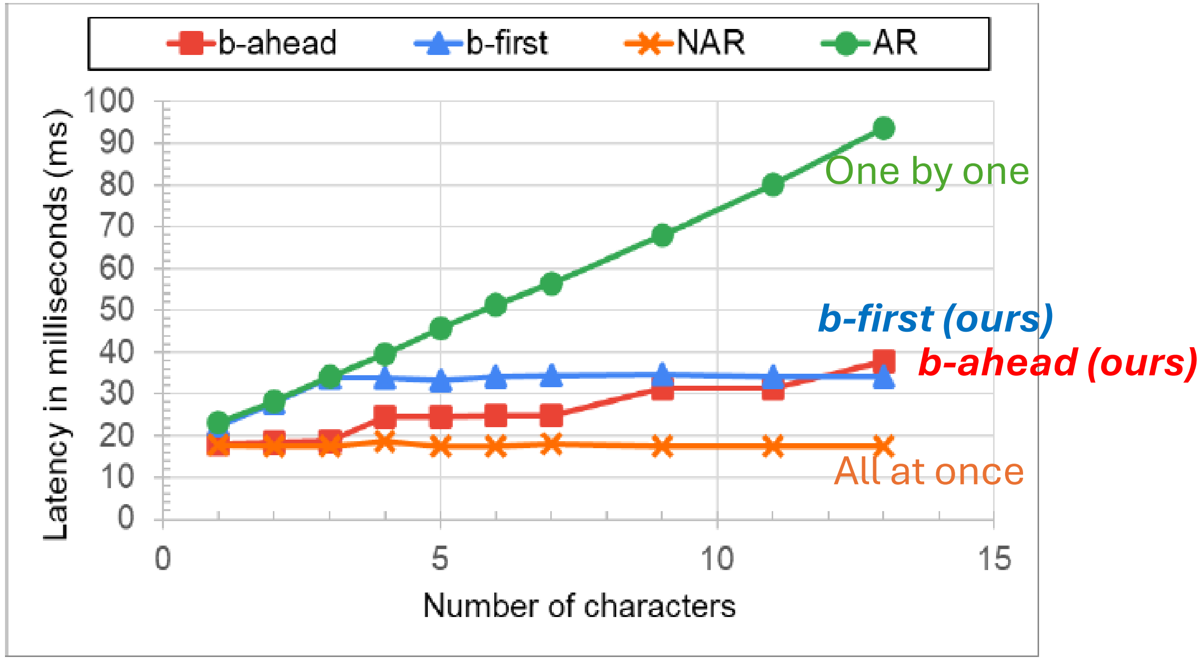
Accuracy ↑ (%) comparisons – at different *b*



RQ2: Partially Autoregressive Decoder for STR
Experimental setup & efficiency comparison

- **Efficiency analysis :**
  - **significant reduction** of **latency** with a marginal accuracy loss (<0.5%).

Latency↓ comparison



RQ2: Partially Autoregressive Decoder for STR
Accuracy comparisons with the recent existing methods

- Comparing with the most **recent SOTA methods** decoding **one by one**:
  - **Consistently improving accuracy** despite having **fewer decoding steps.**
- Thus, the **proposed decoding strategies** can effectively reduce number of decoding steps to b or by b while **maintaining high accuracy**.

Accuracy ↑ (%)comparison with the recent existing methods

Method	IIIT	SVT	IC13	IC15	SVTP	CUTE	Total	N
TRBA* [95]	94.8	91.3	94.0	80.6	82.7	88.1	89.6	n
DiG-ViT-T [23]	96.4	94.4	96.2	87.4	90.2	94.1	93.4	n
DiG-ViT-S [23]	97.7	96.1	97.3	88.6	91.6	96.2	94.7	n
DiG-ViT-B [23]	97.6	96.5	97.6	88.9	92.9	96.5	94.9	n
TRBA* [15]	**98.6**	97.0	97.6	89.8	93.7	97.7	95.7	n
MAERec (no pre-training) [113]	97.4	95.7	97.3	86.7	91.0	96.2	94.1	n
MAERec (pre-training) [113]	98.0	96.8	97.6	87.1	93.2	**97.9**	95.1	n
PARSTR-3-First (Ours)	97.7	97.5	98.0	90.5	95.0	97.2	95.7	b
PARSTR-4-First (Ours)	97.8	97.4	98.0	**90.7**	95.0	97.2	95.8	b
PARSTR-5-First (Ours)	97.8	97.5	98.0	**90.7**	**95.2**	97.6	95.8	b
PARSTR-3-Ahead (Ours)	98.4	97.4	**98.4**	90.3	94.9	97.9	96.0	n/b
PARSTR-4-Ahead (Ours)	98.3	97.4	98.2	90.4	94.9	97.2	95.9	n/b
PARSTR-5-Ahead (Ours)	98.0	96.9	98.0	90.2	94.7	96.9	95.7	n/b

RQ2: Partially Autoregressive Decoder for STR
Known limitations

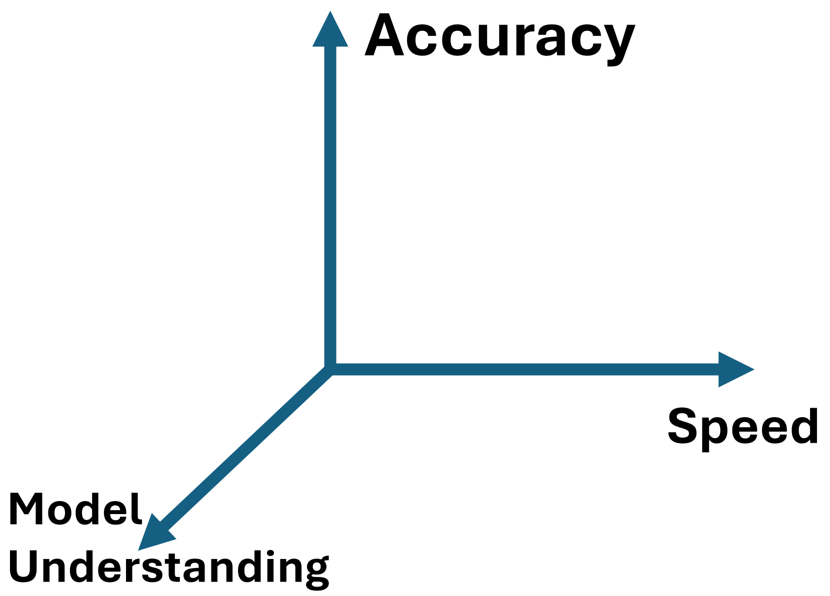
- The proposed decoding schemes:
  - Fixed choice of b - not adaptive
  - b-first & b-ahead: independent decoding, no dynamic selection

Concluding Remarks

Conclusions

Text Recognition – Constrained Multi-Objective Optimization

- Design considerations:
  - **Accuracy**
  - **Complexity**
  - **Explainability**
  - Others
- Constraints:
  - **Script structure** (Latin vs. non-Latin)
  - **Text modality** (word vs. textline; regular vs. curved)
  - **Deployment resource**
- Thus, requiring innovative techniques to **enhance model explainability, reduce complexity,** and **maximize accuracy.**



Conclusions

Summary of Findings

- In the **RQ1**, to **reduce model complexity**, we introduced a **text region selection** technique, which can:
  - **reduce latency** by **half** with a marginal accuracy loss.
  - **obtain comparable** and **better accuracy** with the **high-complexity SOTA** methods.
- In the **RQ2**, to **speed up decoding** process, we introduced **two innovative decoding approaches** (**b-first & b-ahead**) and a **PAR decoder**, which can:
  - **reduce latency** up to **5 times** and to **at most 5 steps** with accuracy loss of <0.5%.
  - **obtain comparable** and **better accuracy** with the **SOTA methods decoding** one by one.

Conclusions
Way Forward

1. Addressing the limitations of each proposed techniques.
2. Novel combinations of these techniques.

Thank you!