Visual Document Understanding with Small VLMs

Visual-NLP Team, John Snow Labs

Agenda

Main topic	Introduced Concepts
Introduction	Definitions for VLMs. Architectures' evolution.
Motivation	VLMs vs. Traditional ML, vs. LLMs. Visual Document Understanding.
VLMs in practice	Prompting vs. Using Task Specific models. Different Output formats. Advantages & disadvantages.
Prompting Small VLMs	Practical Advice.
Handwritten Text Extraction	Example of models on real documents.
Form Extraction	Task Definition & Examples. Hierarchical Document Parser.
General Purpose Models	CPU & GPU models that handle general document processing tasks.
Questions and Answers	Engage in discussions with the presenter.

Presenters today

Alberto

What are Small VLMs?

As of 2025, a **“small” VLM** typically means:

- **Under 10B parameters**.
- Fine-tuned on **limited multimodal datasets.**
- **No long-context image reasoning**, limited to 1–2 images.
- **Weaker OCR and world-knowledge grounding.**

Small VLM evolution

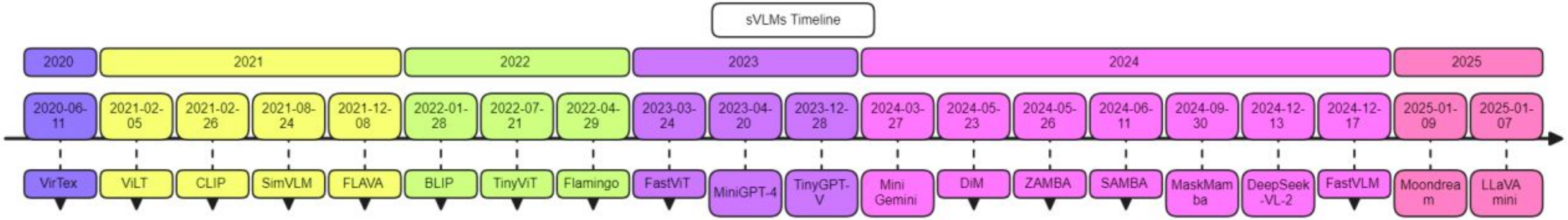


Figure 3: Evolution of Vision Language Models over time

Motivation

Model Type	Input Modality	Output	Typical Use	Flexibility
Traditional CV	Images only	Labels, boxes	Classification, detection	Low, Fixed Task.
Small VLMs	Images + Text	Free text, structured data	Form Parsing, OCR reasoning, DocVQA.	Intermediate. Prompting is possible.
Large Multimodal LLMs	Any media	Complex reasoning	Dialogue, code, reasoning, multimodal understanding	Highest, complex reasoning.

Motivation

Compared to LLMs		Compared to Traditional ML/CV	
Advantages	**Disadvantages**	**Advantages**	**Disadvantages**
Runs on consumer GPUs (8–24 GB VRAM) — you can deploy locally. Runs on CPU, slower but still useful for any use cases. Fine-tunable for niche tasks with small datasets (e.g., invoice parsing).	Poor long-context reasoning (e.g., interpreting 3 charts + a paragraph together) Limited visual resolution understanding — can miss small details.. Few-shot performance drops quickly when task deviates from training domain.	Can adapt to changes in task definition. Can solve more than a single task. **VLMs collapse task-specific pipelines into language prompts.**	Can be heavier to run requiring more memory and compute. It’s hard to get image coordinates. Prompting is not free.

Visual Document Understanding

Tasks we care about,

1. OCR.
2. Form Parsing.
3. Document Structure Recognition/Parsing.
4. DocVQA.

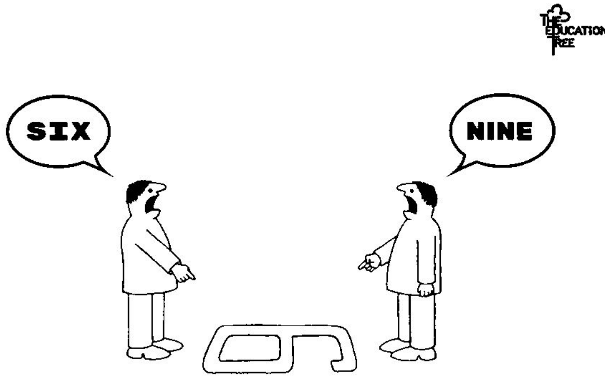
VLMs in Practice

We find them in two flavors,

Task Specific: optimized to solve one problem parsing entire documents, extracting patient information, parsing tables.

General Purpose: accepts a user prompt so that the task(s) can be adjusted according to use preferences.

Representation



- But it's just text!
- You can have different representations for the same page.
- Problem's origin: 2D page vs. 1D char stream.
- The more detail you capture, the less like the original document the output will look.
- This is something you will face in practice.

This is not good :(

```yaml
texts:
#...
- self_ref: '#/texts/5'
  orig: "Let's swim!"
  text: "Let's swim!"
  label: section_header
  level: 1
  children:
  - $ref: '#/texts/6'  # text: To get started with swimming, first ...
  - $ref: '#/groups/0'
  - $ref: '#/texts/10' # text: Also, don't forget:
  - $ref: '#/groups/1'
  - $ref: '#/texts/14' # text: Hmm, what else?
  #...
  parent:
    $ref: '#/texts/1'
  prov: []

groups:
- self_ref: '#/groups/0' # This is a container for list items
  name: list
  label: list
  children:
  - $ref: '#/texts/7' # list_item: You can relax and look around
  - $ref: '#/texts/8' # list_item: Paddle about
  - $ref: '#/texts/9' # list_item: Enjoy summer warmth
  parent:
    $ref: '#/texts/5'
- self_ref: '#/groups/1' # This is a container for list items
  name: list
  label: list
  children:
  - $ref: '#/texts/11' # list_item: Wear sunglasses
  - $ref: '#/texts/12' # list_item: Don't forget to drink water
  - $ref: '#/texts/13' # list_item: Use sun cream
  parent:
    $ref: '#/texts/5'
```

Let’s swim!

To get started with swimming, first lay down in a water and try not to drown:

- You can relax and look around
- Paddle about
- Enjoy summer warmth

Also, don’t forget:

1. Wear sunglasses
2. Don’t forget to drink water
3. Use sun cream

Hmm, what else…

Prompting #1

Small VLMs don't infer the task automatically — **state it clearly**.

```
You are an assistant that describes images in detail. Describe the
image focusing on the objects, their colors, and relative positions.
```

Prompting #2

If the task is composite (e.g., detect + reason + output structured data), break it down:

```
Step 1: Identify all visible text in the image.
Step 2: Describe the layout and objects.
Step 3: Summarize the main activity or purpose.
```

This scaffolding helps smaller models structure their limited reasoning bandwidth.

Prompting #3

Ask for Structured Outputs

Guide the model's response format to reduce confusion:

```
Describe the image and output as JSON:
{
  "Patient Name": “Sarah Smith”,
  "Age": 82,
  "description": "The patient is a pleasant 82yo with a history of.."
}
```

Prompting #4

Avoid vague prompts like *“What’s happening here?”*.
Instead, point attention:

```
Focus on the upper-left part of the image. Return any patient
information in JSON format, respect the following schema,

{
  "Patient Name": “Sarah Smith”,
  "Age": 82,
  "description": "The patient is a pleasant 82yo with a history
of.."
}
```