Computer Vision - 2025

Week #12. Multi-Modal Data Processing. Vision-Language Models

Lectures by Alexei Kornaev [1,2,3]
Practical sessions by Kirill Yakovlev [2]

[1]AI Institute, Innopolis University (IU), Innopolis
[2]Robotics & CV Master's Program, IU, Innopolis
[3]Dept. of M^2R, Orel State University, Orel

[4]RC for AI, National RC for Oncology, Moscow

April 7, 2025

Agenda

1. Introduction
2. Outcomes
3. CV has since teamed up with NLP
4. Contrastive Language-Image Pre-training (CLIP)
5. Other Types of Multimodalities

Section 1. Introduction

Introduction to Multimodality

What is Multimodality?

- Learning from and aligning **heterogeneous data types** (e.g., images, text, sensors, actions).
- Core challenge: Bridging semantic gaps between modalities (e.g., pixels <-> words).

Why It Matters for Robotics

- Robots operate in multimodal environments (sight, sound, language, touch).
- Enables **natural interaction**: "Pick up the blue block" requires aligning vision + language.
- Critical for generalization beyond rigid, pre-programmed tasks.

Recap: Contrastive Loss and Temperature Tuning

Contrastive Loss Fundamentals

For a batch of N image-text pairs: **Goal**: Align positive pairs (I_i, T_i), repel negatives $(I_i, T_{j\neq i})$. **Similarity**: $s_{ij} = \cos\_\text{sim}(I_i, T_j) \in [-1, 1]$. **Batch Scaling**: Loss improves with larger N (more negatives).

Property	Contrastive Loss	Triplet Loss
Negatives per pair	$N-1$	1 (per anchor-positive)
Gradient Signal	All negatives	Margin-based
Modality Support	Cross-modal (e.g., image <-> text)	Single-modality

Table: Contrastive vs. Triplet Loss

Temperature (τ) tuning (softmax scaling):

$$p(I_i, T_j) = \frac{e^{s_{ij}/\tau}}{\sum_k e^{s_{ik}/\tau}}.$$

Low τ (e.g., 0.01) $\rightarrow$ Focuses on hardest negatives. **High** τ (e.g., 1.0) $\rightarrow$ Treats all negatives equally.

Core Concepts in Multimodal Learning

Key Technical Challenges

- **Embedding Alignment**: Map modalities to a shared space (e.g., CLIP's image/text encoders).
- **Cross-Modal Attention**: Dynamically fuse modalities (e.g., Flamingo's Perceiver Resampler).
- **Scaling Laws**: Training with massive datasets (LAION-5B, RT-1).

Contrastive Learning Formulation

$$\mathcal{L}_{\text{contrast}} = -\log \frac{e^{s(I,T)/\tau}}{\sum_{j=1}^{N} e^{s(I,T_j)/\tau}}$$

- $s(I, T)$: Cosine similarity between image I and text T.
- τ: Temperature parameter (learned in CLIP).

Cross-Modal Attention

$$\text{Attention}(Q_{\text{text}}, K_{\text{image}}, V_{\text{image}}) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V$$

Modern VLMs and Applications

Architectures

- Flamingo [Alayrac, 2022]: Processes interleaved images/text for few-shot learning.
- LLaVA [Liu, 2023]: Connects vision encoder to LLM via projection layers.
- BLIP-2 [Li, 2023]: Q-Former bridges frozen encoders (ViT + LLM).

Robotics Applications

- **PALM-E** [Driess, 2023]: Embodied LLM for planning with vision-language-action.
- RT-2: VLMs for robotic control ("pick up the banana").
- **Instruction Following**: Grounding language commands to sensorimotor actions.

Modality	**Robot Input**	**Embedding Technique**
Vision	Camera frames	ViT/ResNet
Language	Commands	BERT/GPT
Actions	Joint angles	MLP

Table: Multimodal Inputs in Robotics

Introduction

Outcomes

CV has since teamed up with NLP

Contrastive Language-Image Pre-training (CLIP)

Other Types of Multi-modalities

Section 2. Outcomes

Outcomes

This week's lecture on Multimodal Data Processing introduces foundational concepts in vision-language alignment for robotic systems. By the end of this session, students will be able to:

1. Explain contrastive learning principles and cross-modal attention mechanisms in Vision-Language Models (VLMs).
2. Implement zero-shot inference using CLIP for robotic object recognition and scene understanding.
3. Critically evaluate architectural choices in modern VLMs (e.g., Flamingo [Alayrac, 2022], LLaVA [Liu, 2023]).

Key Takeaway: Multimodal alignment bridges perception (vision) and reasoning (language), forming the foundation for embodied AI systems like PALM-E [Driess, 2023] in robotics.

Section 3. CV has since teamed up with NLP

CLIP: Contrastive Language-Image Pretraining

Key Components

- Dual-encoder architecture: Image (ViT/ResNet) + Text (Transformer)
- Contrastive learning objective: Align image-text pairs
- Zero-shot transfer via text prompts [Radford et al., 2021]

Contrastive Loss

$$\mathcal{L}_{\text{CLIP}} = -\frac{1}{2N}\sum_{i=1}^{N}\left[\log\frac{e^{s_{ii}/\tau}}{\sum_j e^{s_{ij}/\tau}} + \log\frac{e^{s_{ii}/\tau}}{\sum_j e^{s_{ji}/\tau}}\right]$$

where $s_{ij} = \text{cos\_sim}(I_i, T_j)$

Encoder	Architecture	Dim
Image	ViT-B/16	512
Text	Transformer	512

Table: CLIP Architecture Specifications

Vision-Language Models (VLMs)

Architecture Types

- Dual-Encoder: Fast retrieval (CLIP)
- Fusion-Encoder: Cross-attention (ViLBERT)
- Single-Stream: Unified processing (VisualBERT)

Cross-Modal Attention

$$\text{Attention}(Q_{\text{text}}, K_{\text{image}}, V_{\text{image}}) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$$

Training Objectives

- Image-Text Matching (ITM)
- Masked Language Modeling (MLM)
- Contrastive Loss

VLAM: Vision-Language-Action Models

Key Components

- Multimodal encoder (vision + language)
- Policy network for action generation
- Integration with reinforcement learning [Driess, 2023]

Policy Gradient Theorem

$$\nabla_\theta \mathcal{J}(\theta) = \mathbb{E}_{\pi_\theta} \left[\nabla_\theta \log \pi_\theta(a|s) \cdot A(s, a) \right]$$

Component	Implementation
Multimodal Encoder	Transformer Fusion
Policy Network	MLP/Transformer Decoder
Action Space	Continuous (RL) / Discrete (IL)

Table: VLAM Architecture Components

LLM Training: From Scratch vs Pretrained

Decision Factors

- **Pretrained**: 99% of use cases (low-resource adaptation)
- **From Scratch**: Specialized domains, novel architectures

Parameter-Efficient Fine-Tuning

- LoRA: $\Delta W = BA$ where $B \in \mathbb{R}^{d \times r}$, $A \in \mathbb{R}^{r \times k}$
- QLoRA: 4-bit quantization + LoRA

Metric	From Scratch	Pretrained
Data Needs	1B+ tokens	1k-100k tokens
Compute Cost	$100k+	$100-$1k
Training Time	Weeks	Hours

Table: Training Strategy Comparison

Section 4. Contrastive Language-Image Pre-training (CLIP)

CLIP Loss: Core Mechanism

Definition

Align image-text pairs in a shared space using symmetric contrastive loss:

$$\mathcal{L}_{\text{CLIP}} = -\frac{1}{2N} \sum_{i=1}^{N} \left[\log \frac{e^{s_{ii}}/\tau}{\sum_j e^{s_{ij}}/\tau} + \log \frac{e^{s_{ii}}/\tau}{\sum_j e^{s_{ji}}/\tau} \right]$$

where $s_{ij} = \text{cos\_sim}(I_i, T_j)$ for image and text embeddings, τ is the temperature parameter (learned or fixed) to scale logits.

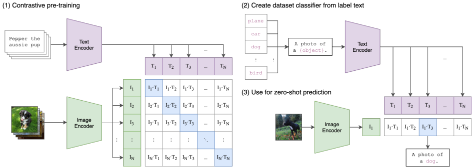

Figure: Contrastive Language-Image Pre-training (CLIP) [Radford et al., 2021]

CLIP Loss: an example

Example: Batch of 2 Image-Text Pairs

Pairs: (Dog Image, "Dog"), (Cat Image, "Cat")
Similarity Matrix:

$$S = \begin{bmatrix} 1.0 & 0.2 \\ 0.1 & 0.9 \end{bmatrix} \quad (\tau = 0.07)$$

Image → Text Loss for Dog Image:

$$-\log \frac{e^{1.0/0.07}}{e^{1.0/0.07} + e^{0.2/0.07}} \approx -\log \frac{e^{14.28}}{e^{14.28} + e^{2.85}} \approx 0$$

Text → Image Loss for "Cat":

$$-\log \frac{e^{0.9/0.07}}{e^{0.1/0.07} + e^{0.9/0.07}} \approx -\log \frac{e^{12.85}}{e^{1.42} + e^{12.85}} \approx 0$$

CLIP Loss vs. Cross-Entropy (CE) Loss

Key Differences

Property	**Standard CE**	**CLIP Loss**
Classes	Fixed (e.g., 1000 ImageNet labels)	Dynamic (batch-paired texts/images)
Negatives	Implicit (non-target classes)	Explicit (all non-diagonal pairs)
Modality	Single (e.g., image $\rightarrow$label)	Cross-modal (image <->text)
Temperature (τ)	Fixed or tuned (usually 1.0)	Learned (e.g., 0.07)

Example: CE for Image Classification

Task: Classify dog/cat images.
Logits: $[2.0, 0.5]$ (dog=2.0, cat=0.5)
CE Loss: $-\log \frac{e^{2.0}}{e^{2.0}+e^{0.5}} \approx 0.12$. Gues what is the label here?

CLIP Loss vs CE Analogy

- CLIP treats each image/text pair as a unique "class".
- CE loss for CLIP is computed over dynamic in-batch negatives.

CLIP vs. Triplet Loss

Triplet Loss Formulation

For anchor a, positive p, negative n:

$$\mathcal{L}_{\text{triplet}} = \max(0, s(a, n) - s(a, p) + \text{margin})$$

Example Comparison

- **Triplet Loss**: Requires explicit triplets (anchor, positive, negative).
  Example: Anchor=Dog Image, Positive="Dog", Negative="Cat".
- **CLIP Loss**: Uses all non-diagonal pairs as negatives.
  Example: For Dog Image, all texts except "Dog" are negatives.

Property	**Triplet Loss**	**CLIP Loss**
Negatives per sample	1	$N - 1$ (batch size - 1)
Training efficiency	Low (needs triplets)	High (batch-level)
Modality support	Single/cross-modal	Cross-modal

Table: Triplet vs CLIP Loss

Practical Insights for CLIP Loss

Hyperparameter Sensitivity

- **Batch Size**: Larger N -> better performance (more negatives). CLIP used $N = 32,768$
- **Temperature (τ)**: Controls "peakiness" of softmax. Too high -> underfitting; too low -> overconfidence.

Example: Impact of τ

For $s(I, T) = 1.0$ and $s(I, T_{\text{neg}}) = 0.2$:

$$\tau = 0.07 : \frac{e^{14.28}}{e^{14.28} + e^{2.85}} \approx 0.999$$

$$\tau = 1.0 : \frac{e^{1.0}}{e^{1.0} + e^{0.2}} \approx 0.67$$

Smaller τ amplifies differences between positive/negative.

CLIP Loss Example: Small Batch (N=2)

Setup

Batch size $N = 2$, temperature $\tau = 0.07$; **image embeddings**: $I_1 = [0.8, 0.6]$, $I_2 = [0.6, 0.8]$; **iext embeddings**: $T_1 = [0.7, 0.7]$, $T_2 = [0.7, -0.7]$; normalized embeddings: $\|I\| = \|T\| = 1$; **similarity Matrix**:

$$S = \begin{bmatrix} I_1 \cdot T_1 & I_1 \cdot T_2 \\ I_2 \cdot T_1 & I_2 \cdot T_2 \end{bmatrix} = \begin{bmatrix} 0.98 & 0.14 \\ 0.98 & -0.14 \end{bmatrix}.$$

Step 1: Image $\to$ Text Loss (for I_1)

$$\text{Softmax}_\tau(S_{I_1}) = \frac{e^{0.98/0.07}}{e^{0.98/0.07} + e^{0.14/0.07}} = \frac{e^{14}}{e^{14} + e^{2}} \approx 1.0,$$

$$\mathcal{L}_{CE}(I_1) = -\log(1.0) \approx 0.$$

Step 2: Text$\to$ Image Loss (for T_2)

$$\text{Softmax}_\tau(S_{T_2}) = \frac{e^{-0.14/0.07}}{e^{0.14/0.07} + e^{-0.14/0.07}} = \frac{e^{-2}}{e^{2} + e^{-2}} \approx 0.018,$$

$$\mathcal{L}_{CE}(T_2) = -\log(0.018) \approx 4.0.$$

Total Loss

$$\mathcal{L}_{\text{CLIP}} = \frac{1}{2 \times 2}(0 + 4.0 + \dots) \quad \text{(Sum over all pairs)}.$$

Temperature Scaling (τ) Demonstration

Introduction

Outcomes

CV has since teamed up with NLP

Contrastive Language-Image Pre-training (CLIP)

Other Types of Multi-modalities

Same Similarities, Different τ

$$S = \begin{bmatrix} 0.98 & 0.14 \\ 0.98 & -0.14 \end{bmatrix}$$

$\tau = 0.07$ (CLIP Default)

$$\text{Softmax}(S_{I_1}) = [0.999, 0.001]$$

Sharp distribution: Focuses on hardest negatives.

$\tau = 1.0$

$$\text{Softmax}(S_{I_1}) = [0.67, 0.33]$$

Softer distribution: Treats negatives more equally.

Implications

- Low τ: Good for clean data, high confidence pairs.
- High τ: Robust to noisy/crowded embedding spaces.

Large-Batch Effect (CLIP-Scale Training)

CLIP Original Training

- Batch size $N = 32,768$
- Each image/text paired with 32k negatives.
- Requires massive compute (thousands of GPUs).

Example: $N = 4$ (Small Scale)

$$S = \begin{bmatrix} 0.98 & 0.14 & 0.2 & 0.1 \\ 0.98 & -0.14 & 0.3 & 0.4 \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

- Hard negatives (e.g., $S_{1,3} = 0.2$) dominate learning.

Why Large Batches Help

- More negatives -> better estimate of true distribution.
- Exposes model to diverse failure modes.

CLIP Loss: Symmetric CE for Contrastive Learning

Core Idea CLIP loss is the sum of two cross-entropy (CE) losses:

$$\mathcal{L}_{\text{CLIP}} = \frac{1}{2}\left(\mathcal{L}_{\text{CE}}^{\text{image} \to \text{text}} + \mathcal{L}_{\text{CE}}^{\text{text} \to \text{image}}\right).$$

Mathematical Formulation For a batch of N pairs:

- **Image $\to$ Text CE**:

$$\mathcal{L}_{\text{CE}}^{\text{image} \to \text{text}} = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s(I_i, T_i)/\tau}}{\sum_j e^{s(I_i, T_j)/\tau}}.$$

- **Text $\to$ Image CE**:

$$\mathcal{L}_{\text{CE}}^{\text{text} \to \text{image}} = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s(T_i, I_i)/\tau}}{\sum_j e^{s(T_j, I_i)/\tau}}.$$

Why Sum Both Directions?

- Avoids modality collapse (e.g., images dominating text).
- Enables zero-shot queries in both directions (imageâtext and textâimage).

Paper reading

Learning Transferable Visual Models From Natural Language Supervision

State-of-the-art computer vision systems are trained to predict a fixed set of predetermined object categories. This restricted form of supervision limits their generality and usability since additional labeled data is needed to specify any other visual concept. Learning directly from raw text about images is a promising alternative which leverages a much broader source of supervision. We demonstrate that the simple pre-training task of predicting which caption goes with which image is an efficient and scalable way to learn SOTA image representations from scratch on a dataset of 400 million (image, text) pairs collected from the internet. After pre-training, natural language is used to reference learned visual concepts (or describe new ones) enabling zero-shot transfer of the model to downstream tasks. We study the performance of this approach by benchmarking on over 30 different existing computer vision datasets, spanning tasks such as OCR, action recognition in videos, geo-localization, and many types of fine-grained object classification. The model transfers non-trivially to most tasks and is often competitive with a fully supervised baseline without the need for any dataset specific training. For instance, we match the accuracy of the original ResNet-50 on ImageNet zero-shot without needing to use any of the 1.28 million training examples it was trained on. We release our code and pre-trained model weights at this https URL. [Radford et al., 2021].

Hands-on Coding with the Inference CLIP models

CLIP and CLIPSeg

- Image + text (proposed classes) + CLIP model = one-shot classification. The code is available via the link #1.
- Image + text (proposed classes) + CLIPSeg model = one-shot semantic segmentation. The code is available via the link #2.

Section 5. Other Types of Multimodalities

Other Types of Multimodalities

The Five Senses Analogy

Sense	Data Modality	ML Example
Vision	Images/Video	CNNs, ViTs
Auditory	Audio/Waveforms	Spectrogram Transformers
Tactile	Pressure/Texture	Tactile Sensors in Robotics
Olfactory	Chemical Sensors	e-Nose Gas Detection
Gustatory	Molecular Data	Flavor Prediction Models

Emerging Sensor Fusion

- LiDAR+RGB: Autonomous vehicles
- IMU+Vision: Human pose estimation
- Spectrograms+Text: Audio captioning

Multimodal Optimization Challenges

Physics-Informed Neural Networks (PINNs)

$$\mathcal{L}_{\text{PINN}} = \underbrace{\lambda_d \|u_\theta(x_i) - u_i\|^2}_{\text{Data Loss}} + \underbrace{\lambda_p \|\mathcal{N}[u_\theta](x_j)\|^2}_{\text{Physics Loss}} + \underbrace{\lambda_r \|\theta\|^2}_{\text{Regularization}}$$

- Multi-objective: Data fitting + PDE residuals [Raissi et al., 2017]
- Loss landscape modality gaps cause training instabilities

Multi-Task Tradeoffs

- Pareto optimality in joint losses
- Gradient conflict quantification:

$$\cos(\nabla_\theta \mathcal{L}_i, \nabla_\theta \mathcal{L}_j) < 0$$

- Solution: Uncertainty weighting [Kendall et al., 2018]

Bibliography

Jean-Baptiste et al. Alayrac. Flamingo: a visual language model for few-shot learning. **NeurIPS**, 2022.

Danny et al. Driess. Palm-e: An embodied multimodal language model. **ICML**, 2023.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In **Proceedings of the IEEE conference on computer vision and pattern recognition**, pages 7482–7491, 2018.

Junnan et al. Li. Blip-2: Bootstrapping vision-language pre-training with frozen image encoders and llms. **arXiv:2301.12597**, 2023.

Haotian et al. Liu. Visual instruction tuning. **NeurIPS**, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020.

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations, 2017. URL https://arxiv.org/abs/1711.10561.