

Neural Networks and Deep Learning

Lecture 14: Multimodal and vision-language models; synthesis of the course

Prof. dr. Daniel Traian PELE

Bucharest University of Economic Studies (ASE), Department of Statistics and Econometrics
Romanian Academy, Institute for Economic Forecasting (IPE)
IDA Institute Digital Assets ▪ AI4EFin ▪ MSCA Digital Finance

Learning objectives

- Explain how a vision transformer turns an image into tokens and where its attention goes, on real satellite images.
- Write the contrastive loss of contrastive language–image pre-training (CLIP), use it as a zero-shot classifier and compare it with a convolutional neural network (CNN) trained on labelled images.
- Describe how vision-language models connect an image encoder to a language model, and test what they can read in charts and tables.
- Synthesise the course: choose a model for a business problem, from logistic regression to foundation models, with the right evaluation.

Abbreviations

ASE	Bucharest University of Economic Studies
AUC	area under the receiver operating characteristic curve
CLIP	contrastive language–image pre-training
CNN	convolutional neural network
ECB	European Central Bank
ESG	environmental, social and governance
GAN	generative adversarial network
GNN	graph neural network
GPT	generative pre-trained transformer
ICBE	International Conference on Business Excellence
InfoNCE	information noise-contrastive estimation loss
IPE	Institute for Economic Forecasting, Romanian Academy
LLaVA	large language and vision assistant
LLM	large language model
LSTM	long short-term memory network
MLP	multilayer perceptron
OCR	optical character recognition
RAG	retrieval-augmented generation
RGB	red, green, blue (colour channels)
RL	reinforcement learning
ROC	receiver operating characteristic
SHAP	Shapley additive explanations
VAE	variational autoencoder
VaR	Value-at-Risk
ViT	vision transformer
VLM	vision-language model
XAI	explainable artificial intelligence

Outline

- Multimodal data in business
- Vision transformers
- CLIP: images and text in one space
- Vision-language models: reading charts and tables
- Synthesis of the course

Images, documents and charts as business data

- **Satellite and aerial images**
  - Crop monitoring for agricultural credit and insurance; land use for environmental, social and governance (ESG) reporting
- **Documents**
  - Scanned contracts, invoices, annual reports: text, tables and layout together
- **Charts**
  - Analysts read price charts; can a model read them, and should it, when the numbers exist?
- **This lecture**
  - CLIP and a vision-language model (VLM), measured on real images

(Radford et al., 2021; Bishop & Bishop, 2024)

Business application: land use for credit, insurance and ESG

- **Agricultural credit**
  - A bank checks that the financed plots are cropland and stay cropland over the season
- **Insurance**
  - Flood and drought claims are compared with images before and after the event
- **ESG reporting**
  - Forest cover and industrial sites around the assets of a portfolio
- **The data problem**
  - Images are free (Sentinel-2 every few days), labels are expensive: hence zero-shot and few-shot models

(Helber et al., 2019)

The data: satellite images, market charts, yield-curve tables

Data	Size	Task
EuroSAT: Sentinel-2 images of Europe, red, green, blue (RGB) channels	27 000 images 64×64 pixels 10 classes	land-use classification: zero-shot, few-shot, full training
S&P 500 daily closes (Yahoo Finance), rendered as chart images	945 windows of 120 days	market regime from the picture vs from the numbers
Euro area AAA yield curve, European Central Bank (ECB)	60 tables rendered as images	can a VLM read a number from a table?

- EuroSAT split 80/20 stratified (seed 20261108): 21 600 training and 5 400 test images.

(Helber et al., 2019)

Outline

- Multimodal data in business
- Vision transformers
- CLIP: images and text in one space
- Vision-language models: reading charts and tables
- Synthesis of the course

A vision transformer reads an image as a sequence of patches

- **In one sentence:** cut the image into 16×16 patches, flatten each patch into a vector, and treat the vectors as words.

$$\mathbf{z}_0 = \left[\mathbf{x}_{\text{cls}};\ \mathbf{E}\mathbf{p}_1;\ \ldots;\ \mathbf{E}\mathbf{p}_{196}\right] + \mathbf{E}_{\text{pos}}, \quad \mathbf{p}_j \in \mathbb{R}^{16\cdot 16\cdot 3}$$

$$\mathbf{z}_\ell = \mathbf{z}_{\ell-1} + \mathrm{MSA}(\mathrm{LN}(\mathbf{z}_{\ell-1})), \quad \mathbf{z}_\ell \leftarrow \mathbf{z}_\ell + \mathrm{MLP}(\mathrm{LN}(\mathbf{z}_\ell))$$

- **where**
  - 224×224 image $= 14 \times 14 = 196$ patches + one class token $= 197$ tokens
  - $\mathbf{E}$ = linear patch embedding (width 768)
  - Multi-head self-attention with 12 heads, 12 layers
  - Layer normalisation
  - Multilayer perceptron (MLP).

- **Vision transformer (ViT)**: no convolution
  - Locality must be learned from data.
- **Hence the need for huge pre-training**: CLIP saw 400 million image–text pairs.

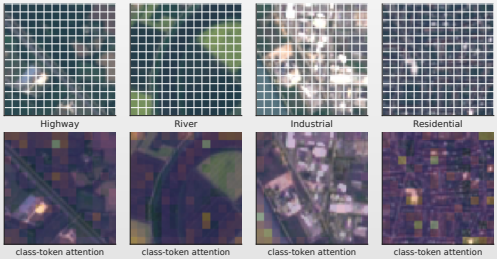
- **In practice:** with few labelled images, a pre-trained ViT beats a ViT trained from scratch by a wide margin.

(Dosovitskiy et al., 2021)

Experiment yourself: notebook §1

Patches and attention on real EuroSAT images

- **In one sentence:** top: the 14×14 patch grid; bottom: where the class token looks, averaged over all layers.



Source: EuroSAT, Sentinel-2 images (Helber et al., 2019). Data: 4 test images upscaled to 224 x 224 pixels = 14 x 14 patches of 16 x 16; model CLIP ViT-B/16.

- Attention rollout: $\tilde{\mathbf{A}} = \prod_\ell (\frac{1}{2}\bar{\mathbf{A}}_\ell + \frac{1}{2}\mathbf{I})$, row of the class token
  - $\bar{\mathbf{A}}_\ell$ = attention of layer ℓ averaged over heads.

(Abnar & Zuidema, 2020)

NN_L14_vit_patches_attention

Outline

- Multimodal data in business
- Vision transformers
- CLIP: images and text in one space
- Vision-language models: reading charts and tables
- Synthesis of the course

CLIP learns by matching images with their captions

- **In one sentence:** in a batch of n image–caption pairs, each image must find its own caption, and vice versa.

$$s_{ij} = \frac{\mathbf{u}_i^\top \mathbf{v}_j}{\tau}, \quad \mathbf{u}_i = \frac{f(I_i)}{\|f(I_i)\|}, \ \mathbf{v}_j = \frac{g(T_j)}{\|g(T_j)\|}$$

$$\mathcal{L} = -\frac{1}{2n} \sum_{i=1}^{n} \left[\log \frac{e^{s_{ii}}}{\sum_j e^{s_{ij}}} + \log \frac{e^{s_{ii}}}{\sum_j e^{s_{ji}}} \right]$$

- **ViT-B/32**: 151.3 million parameters
  - **ViT-B/16**: 149.6 million.
- **No class labels** were used in training: only captions from the web.

- **where**
  - f = image encoder (ViT)
  - g = text encoder (transformer)
  - τ = learned temperature
  - Information noise-contrastive estimation (InfoNCE) loss in both directions.

- **In practice:** one embedding space for images and text enables search, zero-shot labels and retrieval for retrieval-augmented generation (RAG).

(Radford et al., 2021; Oord et al., 2018)

Zero-shot classification with class names as text prompts

- **In one sentence:** embed one sentence per class, embed the image, and pick the closest sentence.

$$\mathbf{w}_c = \mathrm{norm}\left(\frac{1}{M}\sum_{m=1}^{M} g\left(\text{template}_m(c)\right)\right)$$

$$\hat{c}(I) = \arg\max_c \ \mathbf{u}(I)^\top \mathbf{w}_c$$

- **Best zero-shot**: CLIP ViT-B/16, 56.4 % on 5 400 test images, without one labelled example.
- **Chance** with 10 classes: 10 %.

- **where**
  - Template_m = sentence such as "a satellite photo of {class}."
  - $M = 1$ or 5 templates (prompt ensemble)
  - Class descriptions as in Radford et al. (2021).

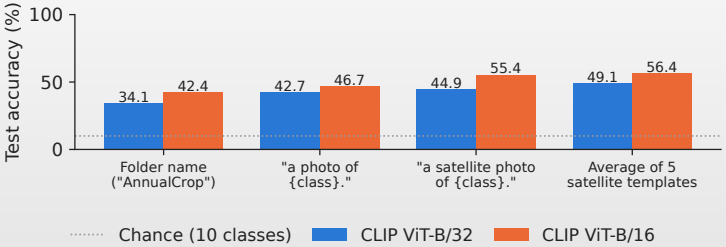
- **In practice:** the weight vectors are written in English, so a domain expert can edit the classifier without retraining.

(Radford et al., 2021)

Experiment yourself: notebook §2

The words in the prompt change zero-shot accuracy

- **In one sentence:** telling CLIP that the image is a satellite photo is worth more than a bigger model.



Source: EuroSAT, Sentinel-2 images (Helber et al., 2019). Data: 5400 test images, no training.

- ViT-B/16: folder name 42.4 %, "a satellite photo of" 55.4 %, 5-template ensemble 56.4 %
  - ViT-B/32 ensemble 49.1 %.

- **In practice:** prompt wording is a hyper-parameter: select it on validation images, then freeze it.

NN_L14_zeroshot_prompts

Zero-shot predictions on eight random test images

- **In one sentence:** green: correct; red: wrong; the percentage is the softmax probability of the chosen class.



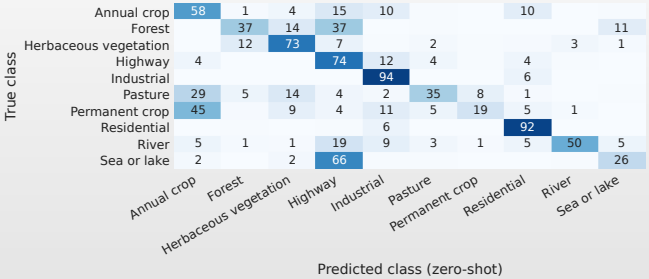
Source: EuroSAT, Sentinel-2 images (Helber et al., 2019). Data: 8 random test images (seeded); green = correct, red = wrong.

- 5 of 8 correct in this sample.

NN_L14_zeroshot_examples

Zero-shot errors on visually similar land uses

- **In one sentence:** rows: true class; columns: predicted class; each row sums to 100 %.



Source: EuroSAT, Sentinel-2 images (Helber et al., 2019). Data: 5400 test images; row percentages.

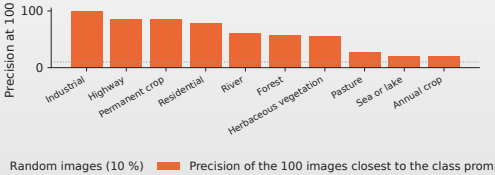
- Best class: industrial (93.6 %)
  - Worst: permanent crop (18.6 %)
  - Largest confusion: 399 of 600 sea or lake images called highway.

NN_L14_zeroshot_confusion

Text-to-image search with the same embeddings

- **In one sentence:** rank all test images by their similarity to a sentence; a bank can find flooded fields or new buildings without labels.

$$\mathrm{P@}k(c) = \frac{1}{k} \sum_{j \in \mathrm{top}_k(\mathbf{w}_c)} \mathbb{1}\{y_j = c\}, \quad \mathrm{top}_k(\mathbf{w}_c) = \text{the } k \text{ images with the largest } \mathbf{u}(I_j)^\top \mathbf{w}_c$$



Source: EuroSAT, Sentinel-2 images (Helber et al., 2019). Data: 5400 test images; model CLIP ViT-B/16, 5 templates.

- Mean precision of the top 100: 58.9 %
  - Best industrial (99.0 %), worst annual crop (20.0 %).

- **In practice:** search first, label the hits, then train a probe: the cheapest way to build a labelled image set.

NN_L14_text_search

A linear probe: CLIP features plus logistic regression

- **In one sentence:** keep CLIP frozen and train only a logistic regression on its image embeddings.

$$\hat{p}(c \mid I) = \mathrm{softmax}\left(\mathbf{W}\mathbf{u}(I) + \mathbf{b}\right)_c$$

$$\min_{\mathbf{W},\mathbf{b}} \; -\sum_i \log \hat{p}(c_i \mid I_i) + \frac{1}{2C}\|\mathbf{W}\|_F^2$$

- **All training images**: 95.5 % (ViT-B/32: 94.8 %).
- **Beats zero-shot** already with one labelled image per class: 59.0 %.

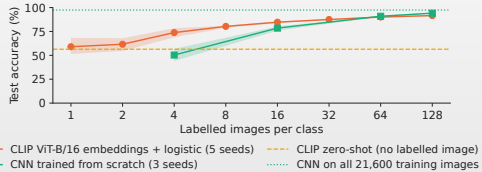
- **where**
  - $\mathbf{u}(I)$ = normalised ViT-B/16 embedding (512 dimensions)
  - $C = 10$ fixed a priori
  - k labelled images per class, 5 seeds.

- **In practice:** a probe trains in seconds on a laptop: the first thing to try when you have a few hundred labels.

Experiment yourself: notebook §3

Pre-trained features and a CNN trained from scratch, by number of labels

- **In one sentence:** one labelled image per class already beats zero-shot; from about 64 per class a small network trained from scratch catches up.



Source: EuroSAT, Sentinel-2 images (Helber et al., 2019). Data: 5400 test images; shaded = range over seeds.

- 16 images per class: probe 84.7 %, convolutional neural network (CNN) 78.5 %
  - 128 per class: 91.6 % vs 94.1 %
  - All 21 600 images: CNN 97.4 %, probe 95.5 %.

- **In practice:** with abundant labels a small CNN catches up
  - With scarce labels, start from a foundation model.

NN_L14_fewshot_eurosat

Size and speed of the three image models

Model	Parameters	ms per image	Test accuracy (%)
CNN from scratch (all labels)	289.2 thousand	–	97.4
CLIP ViT-B/32 + probe	151.3 million	4.23	94.8
CLIP ViT-B/16 + probe	149.6 million	20.49	95.5
Qwen2-VL-2B (chart questions)	2.21 billion	262	–

- Time measured on a laptop (Apple silicon graphics processor) for embedding one image in batches; the VLM time includes the question.
- A bigger patch (32 instead of 16 pixels) means 4 times fewer tokens: faster, slightly less accurate.

(own computation)

Outline

- Multimodal data in business
- Vision transformers
- CLIP: images and text in one space
- Vision-language models: reading charts and tables
- Synthesis of the course

A VLM: an image encoder connected to a language model

- **In one sentence:** image patches are projected into the word-embedding space of a large language model (LLM), which reads them as extra tokens.

$$\mathbf{H}_v = \mathbf{W}_p\, f_{\text{ViT}}(I)$$
$$p(y_t \mid y_{<t}, I, q) = \text{LLM}_\theta\big([\mathbf{H}_v;\ \mathbf{e}(q);\ \mathbf{e}(y_{<t})]\big)$$

- **where**
  - f_{ViT} = vision encoder
  - $\mathbf{W}_p$ = projector (an MLP)
  - q = question
  - Trained first on image–caption pairs, then on visual instructions.

- **Large language and vision assistant (LLaVA)** introduced this recipe
  - Qwen2-VL adds images of any resolution.
- **Our model**: Qwen2-VL-2B-Instruct, 2.21 billion parameters, run locally.

- **In practice:** the answer is generated text: constrain it (yes/no, a number) and score it like a classifier.

(Liu et al., 2023; Wang et al., 2024)

Market regimes as a visual recognition task

- **In one sentence:** draw the last 120 days of the S&P 500 as an image and ask whether the market is in stress, as a chart analyst would.

$$S_t = \mathbb{1}\left\{ \sigma_t^{(20)} > Q_{0.9}\left(\sigma_{t-750:t-1}^{(20)}\right) \right\}$$

$$\sigma_t^{(20)} = \sqrt{252}\,\mathrm{sd}(r_{t-19:t})$$

- **where**
  - S_t = stress label at the end of the window: 20-day volatility above its rolling 90th percentile of the previous three years (past data only)
  - Windows every 5 days
  - Train until 2018, test 394 windows from 2019.

- **Four readers of the same windows**
  - Numbers + logistic regression (returns, volatility, drawdown)
  - CLIP image embedding + logistic regression
  - CLIP zero-shot: "calm, steadily rising" vs "volatile, sharp crash"
  - Qwen2-VL zero-shot: "is the market turbulent?" yes or no

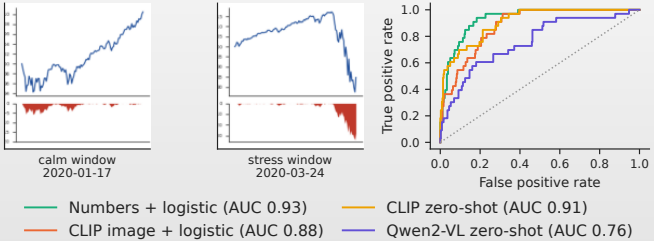
- **In practice:** a rolling-quantile label avoids hand-labelling and look-ahead.

(Jheng, Pele, Tak & Găman, 2026)

Experiment yourself: notebook §4

Detecting market stress from charts and from numbers

- **In one sentence:** left: a calm and a stress window as the models see them; right: receiver operating characteristic (ROC) curves on the test years.



Source: Yahoo Finance, daily quotes. Data: S&P 500, windows of 120 trading days every 5 days; test 7 Jan 2019 to 30 Oct 2026.

- Area under the ROC curve (AUC): numbers 0.933, CLIP probe 0.880, CLIP zero-shot 0.907, Qwen2-VL zero-shot 0.758
  - Stress share in test 8.4 %.

NN_L14_chart_regime

Stress detection: discrimination and precision on the test years

Method	AUC	Avg. prec.	Labels
Numbers + logistic	0.933	0.578	yes
CLIP image + logistic	0.880	0.449	yes
CLIP zero-shot	0.907	0.605	no
Qwen2-VL zero-shot	0.758	0.302	no

- **Best**: numbers.
- **Average precision** of a random score = stress share, 8.4 %.
- **Lesson**: when the numbers exist, a picture adds a lossy step
  - Images earn their place when the numbers are missing (scans, screenshots, satellite).

(own computation, after Jheng, Pele, Tak & Găman, 2026)

Experiment yourself: notebook §4

Reading trends and table cells with a 2-billion-parameter VLM

- **Trend question** on 394 test charts: "did the price end higher or lower?"
  - Accuracy 57.1 % at the 0.5 threshold, below always answering the majority (74.9 %)
  - Yet AUC 0.974: the ranking is right, the threshold is biased — recalibrate on validation charts
- **Table question** on 60 ECB yield-curve tables: "what is the 10-year yield?"
  - Exact answers 25.0 %; mean absolute error 0.192 percentage points; numbers parsed 100.0 %
- **Why it matters**
  - Reading a cell is optical character recognition (OCR) plus locating the column: errors are silent and plausible
  - For documents with known layout, extract the numbers with a parser and use the VLM only where layout varies

(own computation; Wang et al., 2024)

Experiment yourself: notebook §5

The role of a VLM in a bank's document pipeline

- **1. Capture**
  - Scanned page or electronic file; known templates go to a rule-based parser
- **2. Read**
  - OCR and layout for text and tables; a VLM for free-form pages, charts and stamps
- **3. Check**
  - Totals must add up, dates must be valid, values within ranges; a failed check goes to a person
- **4. Measure**
  - Exact-match rate per field on a labelled sample, as for the yield tables (25.0 % here)

Capabilities and limitations of multimodal models

- **Can**
  - Label images with no training data (56.4 % on EuroSAT)
  - Give features that need few labels (84.7 % with 16 per class)
  - Search images by text, and text by images
  - Answer simple questions about charts and documents
- **Cannot (reliably)**
  - Replace the numbers behind a chart
  - Read every cell of a table correctly (25.0 % exact here)
  - Explain faithfully why they answered
  - Separate look-alike classes without examples
- Controls: a labelled test set, constrained answers, human review of low-confidence cases, logs of every prompt and image.

Outline

- Multimodal data in business
- Vision transformers
- CLIP: images and text in one space
- Vision-language models: reading charts and tables
- Synthesis of the course

Methods of the course by data type

Data	Models	Lectures and business cases
Tables	logistic regression, MLP, regularisation	1–3: credit scoring, fraud
Images	CNN, transfer learning, ViT, CLIP	4, 14: land use, documents
Sequences	long short-term memory (LSTM), basis-expansion networks, foundation models	5, 13: prices, load, Value-at-Risk (VaR)
Text	embeddings, transformers, LLM, retrieval-augmented generation (RAG)	6, 7, 12: sentiment, central banks
Generation	variational autoencoder (VAE), generative adversarial network (GAN), diffusion	8: synthetic data, anomalies
Graphs	graph neural network (GNN)	9: transaction networks
Decisions	reinforcement learning (RL)	10: trading, hedging, batteries
Trust	explainable artificial intelligence (XAI): Shapley additive explanations (SHAP), gradients	11: credit audits

Choosing a model for a business problem

- **1. Start with the decision and the cost of errors**
  - A missed fraud and a false alarm do not cost the same; choose the metric before the model
- **2. Fit the benchmark first**
  - Logistic regression, the naive forecast, a Lasso autoregression, a volatility model, word counts with a linear classifier: minutes, interpretable
- **3. Count your labels**
  - Few labels: foundation model zero-shot or with a probe; many labels: a trained network
- **4. Match the architecture to the data**
  - Tables MLP or boosting, images CNN or ViT, sequences recurrent or transformer, graphs GNN
- **5. Prove it**
  - Test once, several seeds, a significance test, calibration, an explanation for the regulator

Five evaluation rules applied throughout the course

- **No leakage**: scalers, prompts and thresholds are chosen on training or validation data
  - Time series are split in time.
- **A benchmark next to every network**: here, a CNN trained from scratch next to CLIP, numbers next to chart images.
- **Seeds and intervals**: the few-shot results above are means over 5 seeds with their range.
- **Metrics for rare events**: average precision and AUC for stress, not accuracy.
- **Real data, reproducible**: every number comes from a script and a public data set, with a Quantlet.

Foundation models for images, text and time series

- **The pattern**
  - Pre-train once on huge unlabelled data; use zero-shot, with a few examples, or with a small trained head
- **Text** (Lecture 12)
  - Prompts and LLM embeddings for sentiment, RAG on ECB speeches, LLM-VaR
- **Time series** (Lecture 13)
  - Chronos-type models forecasting prices, load and VaR without training
- **Images** (this lecture)
  - CLIP zero-shot 56.4 %; probe on all labels 95.5 %
- **The common caveat**
  - Pre-training data may contain your test set; evaluate on data the model cannot have seen

Ten formulas of the course (1/2): learning

Idea	Formula	Lecture
Logistic neuron	$\hat{p} = \sigma(\mathbf{w}^\top \mathbf{x} + b)$	1
Cross-entropy loss	$\mathcal{L} = -\sum_i [y_i \log \hat{p}_i + (1 - y_i) \log(1 - \hat{p}_i)]$	1–2
Backpropagation	$\boldsymbol{\delta}^{(\ell)} = (\mathbf{W}^{(\ell+1)\top} \boldsymbol{\delta}^{(\ell+1)}) \odot f'(\mathbf{z}^{(\ell)})$	2
Gradient step (Adam without moments)	$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}$	2
Weight decay	$\mathcal{L} + \lambda \lVert \boldsymbol{\theta} \rVert_2^2$	3

⊡ All symbols as defined in the lectures cited; σ = logistic function, $\odot$ = element-wise product.

Ten formulas of the course (2/2): architectures

Idea	Formula	Lecture
Convolution	$(\mathbf{K} * \mathbf{X})_{ij} = \sum_{u,v} K_{uv} X_{i+u,j+v}$	4
Recurrent state	$\mathbf{h}_t = \tanh(\mathbf{W}\mathbf{h}_{t-1} + \mathbf{U}\mathbf{x}_t + \mathbf{b})$	5
Attention	$\mathrm{softmax}(\mathbf{Q}\mathbf{K}^\top / \sqrt{d_k})\mathbf{V}$	7, 12–14
Graph convolution	$\mathbf{H}' = \sigma(\tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1/2}\mathbf{H}\mathbf{W})$	9
Contrastive image–text loss	$-\log \frac{e^{s_{ii}}}{\sum_j e^{s_{ij}}}$	14

⊡ $\mathbf{K}$ = kernel; $\tilde{\mathbf{A}}$ = adjacency matrix with self-loops, $\tilde{\mathbf{D}}$ its degree matrix; s_{ij} = scaled image–text similarity.

Responsible use of the models of the course

- **Regulation**
  - The European Union Artificial Intelligence Act (Regulation 2024/1689) lists credit scoring among high-risk uses: documentation, human oversight, data governance
- **Documentation**
  - A model card: data, period, metric, benchmark, seeds, known failure cases (here: sea or lake read as highway)
- **Monitoring**
  - Re-test on new data: price regimes, new satellite seasons, new speech topics
- **Privacy**
  - Local open models keep client documents and images in-house

Requirements for a good course project

- **Question and data**
  - One business decision, one public data set, its period and licence
- **Benchmark first**
  - A linear or naive model, reported next to every network
- **Leakage check**
  - Time-ordered split, scalers fitted on training data, test used once
- **Uncertainty**
  - Several seeds, bootstrap intervals or a Diebold–Mariano test
- **Explanation**
  - SHAP or gradients for the decision that matters
- **Reproducible**
  - One script from raw data to every number and chart

Open research problems for dissertations

- **Calibration under drift**: conformal bands that stay valid when markets change regime.
- **Leakage-proof evaluation** of foundation models trained on the open web.
- **Stable explanations**: do SHAP values of a credit model survive a retraining?
- **Small models for small institutions**: when does a 2–3 billion model on a laptop suffice?
- **Multimodal risk signals**: satellite, text and prices in one early-warning model.
- **Romanian data**: language models and forecasts for Romanian text, markets and energy.

Quiz 1: zero-shot classifier

- A bank wants to flag flooded fields in satellite images but has no labelled flood image
  - What can it do with CLIP?

Answer

- Write class prompts ("a satellite photo of a flooded field", "... of a dry field"), embed them and classify by cosine similarity
  - Then label a few hundred images to validate and train a probe.

Quiz 2: pictures or numbers

- Why did the numeric logistic model detect market stress at least as well as models reading the chart image?

Answer

- The image is drawn from the same numbers, with resolution lost
  - The numbers carry the volatility directly.
  - Images help when no numbers exist.

Quiz 3: labels

- With 16 labelled images per class, which do you choose: a CNN from scratch or a probe on CLIP embeddings?

Answer

- The probe: 84.7 % against 78.5 % for the CNN on our data.

Wrap-up

- A ViT reads images as patch tokens
  - CLIP aligns them with text by a contrastive loss.
- Zero-shot CLIP labels satellite images at 56.4 % with no labels
  - A probe reaches 95.5 %.
- VLMs answer questions about charts and tables, but must be scored like any classifier.
- The course in one line: benchmark first, real data, no leakage, several seeds, a test, an explanation.

Research from our group

- **Multimodal vision-language anomaly detection** (Best Paper Award, International Conference on Business Excellence (ICBE) 2026): market indicators are projected onto multi-panel images, a VLM (GPT-4.1, generative pre-trained transformer) classifies the regime zero-shot with rolling-quantile labels and explains its choice
  - Validated out of sample on the S&P 500 (Jheng, Pele, Tak & Găman, 2026, PICBE).
- **Explainable fraud detection**: tabular models and graph networks combined with SHAP and graph ablation (Diaconu & Pele, 2026, PICBE) — Case study 1.
- **LLM-VaR and LLM-ES**: language models as zero-shot risk models (Pele et al., 2026, Expert Systems with Applications) — Lectures 12 and 13.
- **ESG-aware portfolios**: multi-criteria selection of assets with ESG scores (Găman, Tak, Jheng & Pele, 2026, PICBE).
- Findings as stated in the papers' abstracts; numbers on the other slides come from our own data.

(Bucharest University of Economic Studies, ASE; Romanian Academy, Institute for Economic Forecasting, IPE)

Further reading and resources

- **OpenCLIP** models and benchmarks: github.com/mlfoundations/open_clip.
- **Qwen2-VL**: github.com/QwenLM/Qwen2-VL.
- **Bishop & Bishop (2024)**, chapters 12 and 20: bishopbook.com.
- **Quantinar**: Digital Transformation on Finance: quantinar.com.
- **EuroSAT** data: github.com/phelber/EuroSAT.
- **This lecture**: notebook, seminar and Quantlets — github.com/danpele/NEURAL_BIZ.

References (1/2)

- Abnar, S. & Zuidema, W. (2020). Quantifying attention flow in transformers. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 4190–4197. doi:10.18653/v1/2020.acl-main.385
- Bishop, C. M. & Bishop, H. (2024). *Deep Learning: Foundations and Concepts*. Springer. doi:10.1007/978-3-031-45468-4
- Diaconu, D. & Pele, D. T. (2026). Explainable AI in financial fraud detection: Evidence from the IEEE-CIS fraud dataset. *Proceedings of the International Conference on Business Excellence*, 20(1), 118–132. doi:10.2478/picbe-2026-0015
- Dosovitskiy, A. et al. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations*. arXiv:2010.11929
- Găman, Ș., Tak, R., Jheng, S.-L. & Pele, D. T. (2026). ESG-aware asset selection and portfolio optimization: A comparative study of multi-criteria decision-making methods. *Proceedings of the International Conference on Business Excellence*, 20(1), 133–147. doi:10.2478/picbe-2026-0016
- Helber, P., Bischke, B., Dengel, A. & Borth, D. (2019). EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7), 2217–2226. doi:10.1109/JSTARS.2019.2918242
- Jheng, S.-L., Pele, D. T., Tak, R. & Găman, Ș. (2026). A multimodal vision-language framework for financial anomaly detection. *Proceedings of the International Conference on Business Excellence*, 20(1), 191–201. doi:10.2478/picbe-2026-0020

References (2/2)

- Liu, H., Li, C., Wu, Q. & Lee, Y. J. (2023). Visual instruction tuning. *Advances in Neural Information Processing Systems*, 36. arXiv:2304.08485
- Oord, A. van den, Li, Y. & Vinyals, O. (2018). Representation learning with contrastive predictive coding. arXiv:1807.03748
- Pele, D. T., Bolovăneanu, V., Lin, M.-B., Ren, R., Ginavar, A. T., Spilak, B., Andrei, A.-V., Toma, F.-M., Lessmann, S. & Härdle, W. K. (2026). In the beginning was the Word: LLM-VaR and LLM-ES. *Expert Systems with Applications*, 295, 128676. doi:10.1016/j.eswa.2025.128676
- Radford, A. et al. (2021). Learning transferable visual models from natural language supervision. *Proceedings of the 38th International Conference on Machine Learning*, 8748–8763. arXiv:2103.00020
- Wang, P. et al. (2024). Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. arXiv:2409.12191

Appendix A: the contrastive loss as a classification problem

- Row i of $\mathbf{S} = (s_{ij})$ is a softmax classifier over n captions with the correct class i: the loss is cross-entropy with labels $0, \ldots, n-1$.
- A larger batch means more negatives and a harder task
  - CLIP used $n = 32{,}768$.
- Zero-shot classification is the same computation with the n captions replaced by the C class prompts.

(Radford et al., 2021; Oord et al., 2018)

Appendix B: implementation details

- CLIP: openai/clip-vit-base-patch32 and -patch16 (Hugging Face transformers), images resized to 224 pixels
  - Logistic regression from scikit-learn.
- CNN: three convolution blocks (32, 64, 128 channels), AdamW with a one-cycle schedule, flips and rotations as augmentation
  - 289.2 thousand parameters.
- VLM: Qwen/Qwen2-VL-2B-Instruct, float16
  - Yes/no answers scored by first-token probabilities
  - Numbers by greedy decoding.
- Seed 20261108
  - The full analysis ran in 0.6 minutes on its last run (embeddings and answers cached).