To: M24-RO-01 (master students)

From: Alexei Kornaev, Kirill Yakovlev

Subject: Course on Computer Vision (CV-2025)

Date: May 15, 2025

---

1 Lectures

1. **Intro to Computer Vision** [18, 48, 36, 39, 47, 24, 30]
    - **Theory:** overview of computer vision, it's history, applications, and challenges; image formation; basics of deterministic image processing using filtering, edge detection, and feature extraction; the simplest algorithms of CV for image classification.
    - **Skills:** OpenCV tools for image preprocessing and automatic feature extraction, Pytorch framework, Pytorch Lightning as a tool to simplify the code, and ClearML tool to organize the experiments; comparison of simple algorithms with and without preliminary feature extraction.
2. **Convolutional Neural Networks** [40, 18, 3, 13, 15, 21, 29]
    - **Theory:** Intuition behind convolutions, filters, kernels, and feature maps. Padding, pooling, and striding. Tips and tricks: residual connections, batch normalization. Backpropagation in CNNs and optimization techniques. Architectures: AlexNet, VGG, ResNet, DenseNet, Googlenet, ConvNext, etc. Transfer learning and fine-tuning pre-trained models.
    - **Skills:** Implementation of CNNs using PyTorch on datasets like MNIST and CIFAR-10. Apply transfer learning to adapt pre-trained models (e.g., ResNet, ConvNeXt) to new tasks. Fine-tuning a pre-trained CNN on a custom dataset.
3. **Visual transformers** [40, 51, 11]
    - **Theory:** Transformer architecture: embeddings, positional encoding, self-attention mechanism, multi-head attention, classification block. Vision Transformers (ViTs): patch embeddings, transformer blocks, and classification heads. Comparison of CNNs and ViTs in vision tasks.
    - **Skills:** Implement a Vision Transformer using PyTorch or Hugging Face. Fine-tune ViTs for image classification or segmentation tasks. Compare the performance of ViTs and CNNs on a benchmark dataset.
4. **Assignment # 01**
5. **Segmentation and Object Detection** [40, 44, 4, 49, 19, 22]
    - **Theory:** Semantic segmentation: U-Net, DeepLab, and Mask R-CNN. Object detection: YOLO, Faster R-CNN, and SSD. Instance segmentation and panoptic segmentation. Segment anything model (SAM).

- **Skills:** Implement segmentation models using PyTorch or TensorFlow. Train an object detection model on COCO or Pascal VOC datasets. Perform instance segmentation on a custom dataset.

6. **Maps of Depth and Landmarks Detection** [3, 40, 37, 25]

   - **Theory:** Depth estimation: stereo vision, monocular depth estimation, and LiDAR. Landmark detection: facial landmarks, pose estimation, and keypoint detection.
   - **Skills:** Implement depth estimation using stereo images or monocular methods. Detect facial landmarks using pre-trained models (e.g., dlib or MediaPipe). Build a depth map from a stereo camera setup.

7. **Face Recognition** [54, 31, 50, 23, 45, 10, 52, 40]

   - **Theory:** Face recognition: Traditional methods and deep learning methods. Challenges: pose variation, lighting, occlusion, ethics, adversarial attacks. Architectures (Siamese networks, FaceNet, and ArcFace) and loss functions (contrastive loss, triplet loss, ArcFace loss), and metrics in face recognition.
   - **Skills:** Train a face recognition model using FaceNet or ArcFace. Taking part in CV competitions using Kaggle platform. Implement face detection and recognition in real-time using OpenCV. Build a face recognition system for attendance tracking.

8. **Midterm Exam**

9. **3D Image Processing** [35, 53, 34, 40, 4, 46, 17]

   - **Theory:** 3D reconstruction: structure from motion (SfM) and multi-view stereo. 3D processing: 3D CNNs, 3D Transformers, . CNNs in medical image processing and robotics.
   - **Skills:** Reconstruct 3D models from multiple images using Open3D or PCL. 3D images processing. Hands-on: Build a 3D model from a sequence of images.

10. **Cloud of Points Processing** [3, 41, 27, 26]

    - **Theory:** Point cloud representation: voxelization, octrees, and graph-based methods. Deep learning on point clouds: PointNet, PointNet++, DGCNN, Stratified T.
    - **Skills:** Implement PointNet for point cloud classification. Perform point cloud segmentation using DGCNN or Stratified Transformer. Hands-on: Classify objects in a LiDAR point cloud dataset.

11. **Video Data Processing** [40, 2, 14]

    - **Theory:** Video analysis: optical flow, action recognition, and video summarization. Temporal modeling: RNNs, LSTMs, 3D CNNs, and video transformers (ViViT). Pytorch video library.
    - **Skills:** Extract optical flow from video sequences using OpenCV. Train a 3D CNN for action recognition on UCF101 or Kinetics datasets. Hands-on: Build a video summarization system.

12. **Multimodal Data Processing / Object Tracking on Videos** [42, 1, 32, 28, 12, 40, 7, 33, 43]

    - **Theory (Multimodal Data Processing):** Benefits and key ideas of multimodal learning, contrastive learning, temperature scaling, and cross-modal attention mechanisms in Vision-Language Models (VLMs). Zero-shot inference using CLIP, Flamingo, LLaVA. Multimodal optimization with PINNs.

- **Skills (Object Tracking on Videos):** Implement object tracking using SORT or DeepSORT. Evaluate tracking performance on MOTChallenge datasets. Hands-on: Track multiple objects in a video stream.

13. **Multimodal Data Processing II** [55, 16, 5, 20, 56, 57, 40]

    - **Theory:** DNNs training paradigms. Typology of VLMs: pre-training, transfer, distillation. VLM objectives: contrastive, generative, alignment. Transfer learning for LMs, low-rank adaptation (LoRa). Vision-Language-Action Models (VLAMs): RT, RT-2.
    - **Skills:** Fine-tune CLIP for image-text matching tasks. Hands-on: CLIP + CLIPSeg = Prerequisite for Action.

14. **A lecture of the student's choice**

    - **Approaching Artificial General Intelligence (AGI)** [9]
    - **Dynamic Visual Reasoning by Learning Differentiable Physics** []
    - **Generative Models: VAEs and diffusion models** []
    - **Diving deeper into Vision-Language-Action Models** [6]
    - **Mixing and Tuning of the Models** []

2 Practical sessions

1. **Feature Extraction and Machine Learning** [24, 30]
2. **Convolutional Neural Networks** [24]
3. **Visual Transformers** [24]
4. **Assignment # 01**
5. **Segmentation and Object Detection** [24]
6. **Maps of Depth** [24]
7. **Face recognition** [24]
8. **Midterm Exam**
9. **3D Image Processing** [24]
10. **Cloud of Points Processing** [24, 3]
11. **Video Data Processing** [24]
12. **Object Tracking on Videos** [24]
13. **Multimodal Data Processing: Image and Text classification** [24]
14. **Deploy of a CV model** [24]
15. **Defend of the projects**

3 Assignments

1. **Computer Vision with Real-World Data**
   - **Challenge**: It is human nature to make mistakes. How, then, can the accuracy of the trainable models be improved?
   - **Task**: Design and implement algorithms and models that operate with noisy data (noise in labels, and/or out-of-distribution domain), measure their performance and formulate recommendations on operating with noisy data
   - **Dataset**: CIFAR-10N [38]
   - **Skills**: operating with noisy data, comparison of different loss functions and network architectures, analysis of experimental results
2. **Multimodal Scene Understanding for Robotics**
   - **Challenge**: Robots need to understand their environment using multiple sensors (e.g., cameras, depth sensors) to navigate and interact effectively.
   - **Task**: Create a system that combines RGB images and depth data to perform scene segmentation or object detection for robotic navigation.
   - **Dataset**: Use the **NYU Depth V2** dataset [8], which provides RGB-D images for indoor scenes.
   - **Skills**: Depth estimation, semantic segmentation (e.g., U-Net, DeepLab), and multimodal fusion techniques.

4 Group Project (a paper project, bonus track)

The course project is an opportunity for you to apply the concepts learned in class to a problem aligned with your interests. As this course is part of the **Robotics and Computer Vision Master's Program**, your project should reflect this direction. Projects generally fall into two tracks:

- **Applications**: If you have a background or interest in a specific domain, we encourage you to apply computer vision methods to solve a real-world problem in your field. Identify a practical challenge and address it using **computer vision** and/or **multimodal data processing** techniques.
- **Methods**: You can develop a new model (algorithm) or adapt existing ones to tackle vision or multimodal tasks. This track is more advanced and can potentially lead to publishable work.

This is a **Computer Vision** course, so your project must involve **visual data (pixels)** or **multimodal data** (e.g., combining visual, textual, or sensor data). Projects purely focused on non-visual domains, even if they use convolutional networks, are not suitable.

5 Midterm

A Kaggle competition.

6 Final Exam

A test.

References

[1] J.-B. e. a. Alayrac. Flamingo: a visual language model for few-shot learning. *NeurIPS*, 2022.

[2] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid. Vivit: A video vision transformer, 2021. URL `https://arxiv.org/abs/2103.15691`.

[3] M. Artemyev and A. Ashukha. *Handbook on Machine Learning (in Russian)*. Yandex, 2024. URL `https://education.yandex.ru/handbook/ml`.

[4] C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[5] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control, 2023. URL `https://arxiv.org/abs/2307.15818`.

[6] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control, 2023. URL `https://arxiv.org/abs/2307.15818`.

[7] G. Ciaparrone, F. L. Sánchez, S. Tabik, L. Troiano, R. Tagliaferri, and F. Herrera. Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 381:61–88, 2020.

[8] C. Couprie, C. Farabet, L. Najman, and Y. LeCun. Indoor semantic segmentation using depth information. *arXiv preprint arXiv:1301.3572*, 2013.

[9] DeepSeek-AI, D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, X. Zhang, X. Yu, Y. Wu, Z. F. Wu, Z. Gou, Z. Shao, Z. Li, Z. Gao, A. Liu, B. Xue, B. Wang, B. Wu, B. Feng, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan, D. Dai, D. Chen, D. Ji, E. Li, F. Lin, F. Dai, F. Luo, G. Hao, G. Chen, G. Li, H. Zhang, H. Bao, H. Xu, H. Wang, H. Ding, H. Xin, H. Gao, H. Qu, H. Li, J. Guo, J. Li, J. Wang, J. Chen, J. Yuan, J. Qiu, J. Li, J. L. Cai, J. Ni, J. Liang, J. Chen, K. Dong, K. Hu, K. Gao, K. Guan, K. Huang, K. Yu, L. Wang, L. Zhang, L. Zhao, L. Wang, L. Zhang, L. Xu, L. Xia, M. Zhang, M. Zhang, M. Tang, M. Li, M. Wang, M. Li, N. Tian, P. Huang, P. Zhang, Q. Wang, Q. Chen, Q. Du, R. Ge, R. Zhang, R. Pan, R. Wang, R. J. Chen, R. L. Jin, R. Chen, S. Lu, S. Zhou, S. Chen, S. Ye, S. Wang, S. Yu, S. Zhou, S. Pan, S. S. Li, S. Zhou, S. Wu, S. Ye, T. Yun, T. Pei, T. Sun, T. Wang, W. Zeng, W. Zhao, W. Liu, W. Liang, W. Gao, W. Yu, W. Zhang, W. L. Xiao, W. An, X. Liu, X. Wang, X. Chen, X. Nie, X. Cheng, X. Liu, X. Xie, X. Liu, X. Yang, X. Li, X. Su, X. Lin, X. Q. Li, X. Jin, X. Shen, X. Chen, X. Sun, X. Wang, X. Song, X. Zhou, X. Wang, X. Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. Zhang, Y. Xu, Y. Li, Y. Zhao, Y. Sun, Y. Wang,

Y. Yu, Y. Zhang, Y. Shi, Y. Xiong, Y. He, Y. Piao, Y. Wang, Y. Tan, Y. Ma, Y. Liu, Y. Guo, Y. Ou, Y. Wang, Y. Gong, Y. Zou, Y. He, Y. Xiong, Y. Luo, Y. You, Y. Liu, Y. Zhou, Y. X. Zhu, Y. Xu, Y. Huang, Y. Li, Y. Zheng, Y. Zhu, Y. Ma, Y. Tang, Y. Zha, Y. Yan, Z. Z. Ren, Z. Ren, Z. Sha, Z. Fu, Z. Xu, Z. Xie, Z. Zhang, Z. Hao, Z. Ma, Z. Yan, Z. Wu, Z. Gu, Z. Zhu, Z. Liu, Z. Li, Z. Xie, Z. Song, Z. Pan, Z. Huang, Z. Xu, Z. Zhang, and Z. Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

[10] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. 2019.

[11] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021. URL https://arxiv.org/abs/2010.11929.

[12] D. e. a. Driess. Palm-e: An embodied multimodal language model. *ICML*, 2023.

[13] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[14] B. K. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3): 185–203, 1981.

[15] J. Howard and S. Gugger. *Deep Learning for Coders with Fastai and Pytorch: AI Applications Without a PhD*. O'Reilly Media, Incorporated, 2020. ISBN 9781492045526. URL https://books.google.no/books?id=xd6LxgEACAAJ.

[16] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

[17] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. 2008.

[18] Hugging Face, CV course. Computer vision course by hugging face community. https://huggingface.co/learn/computer-vision-course/unit0/welcome/welcome. Accessed: May 15, 2025.

[19] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma. A review of yolo algorithm developments. *Procedia computer science*, 199:1066–1073, 2022.

[20] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan. Supervised contrastive learning, 2021. URL https://arxiv.org/abs/2004.11362.

[21] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman. 1d convolutional neural networks and applications: A survey. *Mechanical systems and signal processing*, 151: 107398, 2021.

[22] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.

[23] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. 2015.

[24] A. Kornaev and K. Yakovlev. Cv-2025, the course repository. https://github.com/Mechanics-Mechatronics-and-Robotics/CV-2025. Accessed: May 15, 2025.

[25] A. Kumar, T. K. Marks, W. Mou, Y. Wang, M. Jones, A. Cherian, T. Koike-Akino, X. Liu, and C. Feng. Luvli face alignment: Estimating landmarks' location, uncertainty, and visibility likelihood. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8236–8246, 2020.

[26] X. Lai, J. Liu, L. Jiang, L. Wang, H. Zhao, S. Liu, X. Qi, and J. Jia. Stratified transformer for 3d point cloud segmentation, 2022. URL https://arxiv.org/abs/2203.14508.

[27] J. Li, H. Qin, J. Wang, and J. Li. Openstreetmap-based autonomous navigation for the four wheel-legged robot via 3d-lidar and ccd camera. *IEEE Transactions on Industrial Electronics*, 69(3):2708–2717, 2021.

[28] J. e. a. Li. Blip-2: Bootstrapping vision-language pre-training with frozen image encoders and llms. *arXiv:2301.12597*, 2023.

[29] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12):6999–7019, 2021.

[30] Lighthing. Pytorch lightning introduction. a colab notebook. https://colab.research.google.com/drive/1Mowb4NzWlRCxzAFjOIJqUmmk_wAT-XP3. Accessed: May 15, 2025.

[31] C. Liu, K. Hirota, and Y. Dai. Patch attention convolutional vision transformer for facial expression recognition with occlusion. *Information Sciences*, 619:781–794, 2023.

[32] H. e. a. Liu. Visual instruction tuning. *NeurIPS*, 2023.

[33] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, and T.-K. Kim. Multiple object tracking: A literature review. *Artificial intelligence*, 293:103448, 2021.

[34] J. Mao, Y. Xue, M. Niu, H. Bai, J. Feng, X. Liang, H. Xu, and C. Xu. Voxel transformer for 3d object detection, 2021. URL https://arxiv.org/abs/2109.02497.

[35] D. Maturana and S. Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *Ieee International Conference on Intelligent Robots and Systems*, pages 922–928, 2015.

[36] D. Merkulov. Seminar on optimization at mipt. https://mipt21.fmin.xyz/. Accessed: May 15, 2025.

[37] Y. Ming, X. Meng, C. Fan, and H. Yu. Deep learning for monocular depth estimation: A review. *Neurocomputing*, 438:14–33, 2021.

[38] Papers&Code, CIFAR-10N. Learning with noisy labels. https://paperswithcode.com/task/learning-with-noisy-labels. Accessed: May 15, 2025.

[39] S. J. Prince. *Computer Vision: Models, Learning, and Inference*. Cambridge University Press, 2012.

[40] S. J. Prince. *Understanding Deep Learning*. The MIT Press, 2023. URL http://udlbook.com.

[41] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation, 2017. URL https://arxiv.org/abs/1612.00593.

[42] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020.

[43] M. Raissi, P. Perdikaris, and G. E. Karniadakis. Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations, 2017. URL https://arxiv.org/abs/1711.10561.

[44] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. URL https://arxiv.org/abs/1505.04597.

[45] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. 2015.

[46] S. P. Singh, L. Wang, S. Gupta, H. Goli, P. Padmanabhan, and B. Gulyás. 3d deep learning on medical images: a review. *Sensors*, 20(18):5097, 2020.

[47] S. Sridhar. Computer vision, csci 1430, fall 2024. https://browncsci1430.github.io/index.html#schedule-content. Accessed: May 15, 2025.

[48] R. Szeliski. Computer vision. algorithms and applications, 2022. URL https://szeliski.org/Book/.

[49] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *Machine Learning and Knowledge Extraction*, 5(4):1680–1716, 2023.

[50] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1): 71–86, 1991.

[51] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

[52] M. Wang and W. Deng. Deep face recognition: A survey. *Neurocomputing*, 429:215–244, 2021.

[53] P.-S. Wang, Y. Liu, Y.-X. Guo, C.-Y. Sun, and X. Tong. O-cnn: octree-based convolutional neural networks for 3d shape analysis. *ACM Transactions on Graphics*, 36(4):1–11, July 2017. ISSN 1557-7368. doi: 10.1145/3072959.3073608. URL http://dx.doi.org/10.1145/3072959.3073608.

[54] D. Zeng, R. Veldhuis, and L. Spreeuwers. A survey of face recognition techniques under occlusion. *IET biometrics*, 10(6):581–606, 2021.

[55] J. Zhang, J. Huang, S. Jin, and S. Lu. Vision-language models for vision tasks: A survey, 2024. URL https://arxiv.org/abs/2304.00685.

[56] K. Zhou, J. Yang, C. C. Loy, and Z. Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16816–16825, 2022.

[57] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.