модели cv

Отличный запрос! Дополню список более современными, узкоспециализированными и альтернативными архитектурами.
Отличная идея! Вот подробный список ключевых архитектур в Computer Vision, структурированный по типам задач. Это ваш "справочник" для выбора модели.

---

1. Классификация изображений (Image Classification)

LeNet-5 (1998)

- **Основная идея:** Первая практическая сверточная сеть. Последовательность сверток и субдискретизации (пулинга).
- **Плюсы:** Историческое значение, простая архитектура для понимания основ.
- **Минусы:** Устарела для современных задач, малая глубина.
- **Задачи:** Распознавание цифр (MNIST), простые объекты.

AlexNet (2012)

- **Основная идея:** Глубокая CNN (8 слоев), использовала ReLU, Dropout и обучение на нескольких GPU.
- **Плюсы:** Прорыв, доказавший эффективность глубокого обучения на ImageNet.
- **Минусы:** Архитектура менее оптимальна по сравнению с современными, большие полносвязные слои.
- **Задачи:** Общая классификация, исторический baseline.

VGG (VGG16, VGG19) (2014)

- **Основная идея:** Унификация: использует только свертки 3x3 и пулинг 2x2, чтобы наращивать глубину (16-19 слоев).
- **Плюсы:** Очень простая и однородная архитектура, хороший набор предобученных весов для трансферного обучения.
- **Минусы:** Огромное количество параметров (особенно в полносвязных слоях), медленная и "тяжелая".
- **Задачи:** Трансферное обучение, художественный стиль (style transfer), хороша для понимания.

Inception (GoogLeNet) (2014)

- **Основная идея: Inception-модуль**, который выполняет свертки разного размера (1x1, 3x3, 5x5) и пулинг одновременно, а затем конкатенирует результаты. Позволяет эффективно изучать паттерны разных масштабов.
- **Плюсы:** Высокая эффективность при меньшем количестве параметров, чем VGG.
- **Минусы:** Сложная архитектура для самостоятельной реализации.
- **Задачи:** Эффективная классификация, основа для более поздних версий (Inception-v3, v4).

ResNet (Residual Networks) (2015)

- **Основная идея: Остаточные связи (skip connections)**. Вместо обучения прямой функции H(x), сеть учит остаточную функцию F(x) = H(x) - x, где H(x) = F(x) + x. Решает проблему исчезающего градиента.
- **Плюсы:** Позволяет строить очень глубокие сети (50, 101, 152+ слоев), легко обучаются, стали стандартом де-факто.
- **Минусы:** Нет явных минусов для общего применения. Очень популярны.
- **Задачи: Универсальный выбор для классификации, детекции, сегментации в качестве backbone.**

EfficientNet (2019)

- **Основная идея: Сложное масштабирование (compound scaling)**. Сбалансированное масштабирование глубины, ширины и разрешения входного изображения по формуле для оптимального использования ресурсов.
- **Плюсы:** Лучшее соотношение точность/скорость/размер модели среди классических CNN. Много вариантов (B0-B7).
- **Минусы:** Относительно сложная архитектура.
- **Задачи: Идеальный выбор, когда нужен баланс между точностью и эффективностью** (мобильные устройства, продакшн).

Vision Transformer (ViT) (2020)

- **Основная идея:** Применение архитектуры **Transformer** (из NLP) к изображениям. Разбивает изображение на патчи, линейно их проецирует и подает в encoder Transformer.
- **Плюсы:** Превосходит CNN на очень больших датасетах, отличная способность к моделированию глобального контекста.
- **Минусы:** Требует **огромных объемов данных для предобучения**, может быть тяжелее CNN для мелких датасетов без тонкой настройки.
- **Задачи:** Классификация на больших данных, мультимодальные задачи (изображение + текст).

2. Детекция объектов (Object Detection)

R-CNN, Fast R-CNN, Faster R-CNN (2014-2015)

- **Основная идея: Двухстадийные (two-stage) детекторы.**
  1. **Region Proposal:** Генерация регионов-кандидатов (selective search в R-CNN, **Region Proposal Network (RPN)** в Faster R-CNN).
  2. **Классификация и регрессия:** Каждый регион пропускается через CNN для классификации и уточнения bounding box.
- **Плюсы: Высокая точность (особенно для мелких объектов).** Эталон для многих бенчмарков.
- **Минусы:** Относительно **медленные**, сложная тренировочная процедура.
- **Задачи:** Где важна максимальная точность, а скорость не критична (видеонаблюдение, медицинская диагностика).

YOLO (You Only Look Once) (2016 - ...)

- **Основная идея: Одностадийный (one-stage) детектор.** Разбивает изображение на сетку и для каждой ячейки предсказывает bounding boxes и классы **за один проход** сети.
- **Плюсы: Очень высокое быстродействие (real-time).** Простая архитектура.
- **Минусы:** Исторически уступал в точности двухстадийным методам, особенно на мелких объектах (поздние версии сильно улучшили точность).
- **Задачи: Real-time детекция** (автономные автомобили, видеоаналитика, робототехника). **YOLOv8, YOLOv10** — современные SOTA-реализации.

SSD (Single Shot MultiBox Detector) (2016)

- **Основная идея: Одностадийный детектор.** Как YOLO, но использует feature maps **разных масштабов** (с разных слоев сети) для предсказания объектов разного размера.
- **Плюсы:** Хороший баланс скорости и точности (быстрее Faster R-CNN, точнее ранних YOLO).
- **Минусы:** Точность на мелких объектах может страдать.
- **Задачи:** Универсальная детекция, когда нужен баланс скорости и качества.

RetinaNet (2017)

- **Основная идея:** Одностадийный детектор с **Focal Loss**. Focal Loss борется с дисбалансом классов (объек vs. фон), перенаправляя фокус обучения на "сложные"

примеры.
- **Плюсы:** Одностадийная скорость с двухстадийной точностью (на момент выхода).
- **Минусы:** Чуть сложнее и медленнее, чем YOLO/SSD.
- **Задачи:** Детекция с сильным дисбалансом классов, где важна высокая точность.

---

3. Семантическая и Instance сегментация

U-Net (2015)

- **Основная идея: Encoder-Decoder архитектура с skip-connections.**
  - **Encoder (сжатие пути):** Извлекает признаки, уменьшая пространственное разрешение.
  - **Decoder (расширяющий путь):** Восстанавливает карту сегментации, увеличивая разрешение.
  - **Skip connections:** Прямые связи между соответствующими слоями encoder и decoder для сохранения пространственной информации.
- **Плюсы:** Очень эффективна при **небольшом количестве данных** (биомедицинские изображения). Простая и элегантная.
- **Минусы:** Не различает разные экземпляры одного класса (только семантическая сегментация).
- **Задачи: Биомедицинская сегментация**, сегментация спутниковых снимков, где объектов мало, а контекст важен.

Mask R-CNN (2017)

- **Основная идея:** Расширение Faster R-CNN. Добавляет третью ветку — **маленькую сверточную сеть (mask head)**, которая предсказывает бинарную маску для каждого обнаруженного объекта.
- **Плюсы:** Де-факто стандарт для **instance segmentation**. Делает детекцию, классификацию и сегментацию одновременно.
- **Минусы:** Двухстадийный, относительно медленный.
- **Задачи: Instance segmentation** (например, разделение отдельных людей в толпе), где нужны точные маски.

DeepLab (v3+) (2018)

- **Основная идея:** Использование **Atrous (Dilated) Convolutions** и **ASPP (Atrous Spatial Pyramid Pooling)**.

    - **Atrous Convolutions:** Позволяют увеличить **поле восприятия (receptive field)** без потери разрешения.
    - **ASPP:** Применяет свертки с разным темпом дилатации параллельно, чтобы захватывать контекст в разных масштабах.
- **Плюсы:** Высокая точность на границах объектов, эффективное использование контекста.
- **Минусы:** Вычислительно затратнее U-Net.
- **Задачи: Семантическая сегментация** сцен (автономное вождение: дорога, пешеход, знак), где важен контекст и четкость границ.

Краткая шпаргалка по выбору:

- **Нужно просто классифицировать изображение?**
    - Начинайте с **ResNet50** или **EfficientNet-B4**.
    - Если данных очень много и есть ресурсы — пробуйте **ViT**.
- **Нужно найти объекты на изображении (bounding box)?**
    - Для скорости (real-time): **YOLOv8/v10**, **SSD**.
    - Для максимальной точности (не важна скорость): **Faster R-CNN**.
    - Для баланса: **RetinaNet**.
- **Нужно выделить объекты пиксель в пиксель (сегментация)?**
    - Если объекты — это "области", а не отдельные экземпляры (например, небо, трава, вода): **U-Net** (если данных мало), **DeepLabv3+** (если данных много и важны границы).
    - Если нужно разделить каждый экземпляр объекта отдельно (каждый человек, каждую машину): **Mask R-CNN**.
- **Ограничены по вычислительным ресурсам (мобильное устройство)?**
    - **EfficientNet** (классификация), **MobileNet** (как backbone для детекторов/сегментаторов), **YOLO Nano/Tiny** версии.

---

4. Современные и эффективные архитектуры (Backbone)

MobileNet (v1-v3) (2017-2019)

- **Основная идея:** Использование **depthwise separable convolutions** для радикального сокращения вычислений и параметров. **MobileNetV3** добавляет архитектурный поиск

(NAS) и squeeze-and-excitation блоки.
- **Плюсы:** Экстремально легкие и быстрые модели, созданные специально для мобильных и embedded-устройств. Соотношение точность/производительность лучше, чем у старых легких сетей.
- **Минусы:** Точность ниже, чем у "больших" моделей (ResNet, EfficientNet) при равных условиях.
- **Задачи: Мобильные приложения,** IoT, edge-устройства, real-time инференс на ограниченном железе.

DenseNet (2017)

- **Основная идея:** Каждый слой получает на вход **конкатенацию** feature maps всех предыдущих слоев. Плотные соединения (dense connections).
- **Плюсы:** Улучшенный градиентный поток, повторное использование признаков, меньше параметров, чем у ResNet, за счет узких слоев.
- **Минусы:** Высокие требования к памяти (ОЗУ) из-за хранения всех feature maps для конкатенации.
- **Задачи:** Классификация, особенно где важна эффективность использования параметров. Часто используется в медицинских изображениях.

ConvNeXt (2022)

- **Основная идея:** "Омодернивание" классического ResNet с идеями из Transformer'ов (Swin Transformer): большие размеры ядер (7x7), depthwise conv, меньше активационных функций, LayerNorm вместо BatchNorm.
- **Плюсы:** Достигает производительности современных Transformer'ов, сохраняя простоту и вычислительную эффективность чистых сверточных архитектур. Легче для развертывания.
- **Минусы:** Относительно новая, меньше готовых весов и интеграций, чем у ResNet.
- **Задачи: Современная альтернатива ResNet/EfficientNet** для классификации и в качестве backbone. Хороший выбор для продакшена.

Swin Transformer (2021)

- **Основная идея:** Иерархический Vision Transformer с **сдвигаемыми окнами (shifted windows)**. Вычисляет self-attention внутри локальных окон, которые сдвигаются между слоями, что позволяет моделировать как локальные, так и глобальные зависимости, сохраняя линейную вычислительную сложность.
- **Плюсы:** Сочетает преимущества Transformer'ов (глобальный контекст) с индуктивными смещениями CNN (локальность, иерархичность). Эффективен для задач dense prediction (детекция, сегментация).

- **Минусы:** Сложнее CNN, требует больше данных для предобучения, чем ResNet.
- **Задачи: SOTA в детекции и сегментации** (часто используется как backbone в новых методах), где нужен глобальный контекст (например, сцены с множеством взаимодействующих объектов).

---

5. Детекция объектов (продолжение)

DETR (DEtection TRansformer) (2020)

- **Основная идея:** Полностью трансформерный подход к детекции. Прямо предсказывает фиксированный набор bounding boxes, используя **бипартный мэтчинг (bipartite matching)** с ground truth. Не нужны anchor boxes и NMS.
- **Плюсы:** Простейший на уровне кода пайплайн (encoder-decoder трансформер). Глобальные предсказания (избегает артефактов NMS). Лучше справляется с большими объектами.
- **Минусы:** Медленнее обучается, хуже обнаруживает мелкие объекты (по сравнению с оптимизированными CNN методами). Поздние версии (Deformable DETR, DETR-v2) это исправляют.
- **Задачи: Интересный исследовательский подход**, хорош для объектов среднего и крупного размера. Показал, что детекция возможна без анкоров.

YOLO Evolution (v4, v5, v8, v10)

- **Основная идея:** Эволюция оригинальной идеи YOLO. Включает современные трюки: **PANet** (пирамида признаков), **CSPNet** (Cross Stage Partial connections для эффективности), **anchor-free** подход (в v8+), различные активации (SiLU), advanced аугментации (Mosaic, MixUp).
- **Плюсы: Лидер по соотношению скорость/точность в реальном времени.** Невероятно популярна, огромное комьюнити, простой API (особенно Ultralytics YOLOv8/v10). Anchor-free подход упрощает.
- **Минусы:** "Джунгли" версий и реализаций (не все официальные). Внутренняя сложность современной версии высока.
- **Задачи: De facto стандарт для industrial real-time детекции** (логистика, беспилотники, безопасность).

---

6. Сегментация (продолжение)

SegNet (2015) / FCN (Fully Convolutional Network) (2014)

- **Основная идея:** Первые чисто сверточные сети для семантической сегментации. **FCN** заменяет полносвязные слоя на сверточные для выхода карты. **SegNet** использует **индексы пулинга** в декодере для более точного апсемплинга.
- **Плюсы:** Историческое значение, простота. FCN — фундаментальная идея.
- **Минусы:** Результаты грубее, чем у U-Net/DeepLab.
- **Задачи:** Общее понимание эволюции сегментации, простые задачи.

HRNet (High-Resolution Net) (2019)

- **Основная идея:** Сохраняет **высокое разрешение** feature maps на протяжении всей сети, параллельно соединяя ветки высокого и низкого разрешения (многомасштабное слияние).
- **Плюсы:** Отлично сохраняет пространственные детали. **SOTA для задач, критичных к местоположению** (pose estimation, сегментация мелких объектов, landmark detection).
- **Минусы:** Вычислительно затратнее, чем однонаправленные encoder-decoder.
- **Задачи:** Pose estimation (например, COCO Keypoints), детальная сегментация, когда важна точная локализация, а не только семантика.

---

7. Генеративные модели и Representation Learning

Autoencoders (VAE, Denoising AE)

- **Основная идея:** Нейросеть учится сжимать вход (в латентное пространство) и восстанавливать его. **VAE** (Variational AE) учит вероятностное латентное пространство, что позволяет генерировать новые данные.
- **Плюсы:** Могут использоваться для уменьшения размерности, удаления шума, предобучения без учителя, простой генерации.
- **Минусы:** Сгенерированные изображения часто **размытые** (особенно у VAE).
- **Задачи:** Предобработка изображений (denoising, inpainting), простые аномалии детекшн, обучение латентных представлений.

GAN (Generative Adversarial Networks) (2014)

- **Основная идея:** Две сети соревнуются: **Генератор** создает fake-изображения, **Дискриминатор** учится отличать real от fake. В процессе генератор учится создавать реалистичные изображения.
- **Плюсы:** Может генерировать **очень резкие и реалистичные** изображения.

- **Минусы:** Сложно обучать (нестабильность, mode collapse). Современные версии (StyleGAN) очень сложны.
- **Задачи:** Генерация фотореалистичных лиц/объектов, data augmentation, domain translation (day->night), super-resolution (SRGAN).

Diffusion Models (2020-...)

- **Основная идея:** Постепенное добавление шума к данным (forward process), а затем обучение сети обращать этот процесс (reverse process) для генерации данных из шума.
- **Плюсы: SOTA в качестве генерируемых изображений** (превзошли GAN). Более стабильное обучение, чем у GAN. Отличный likelihood.
- **Минусы: Медленный процесс генерации** (требует многих шагов). Ресурсоемкое обучение.
- **Задачи: Современная high-quality генерация** (DALL-E 2, Imagen, Stable Diffusion), inpainting, super-resolution, дизайн.

---

8. Модели для других задач CV

Siamese Networks

- **Основная идея:** Два или более идентичных подсети (с общими весами), которые обрабатывают разные входы для вычисления **схожести** между ними (например, через distance metric).
- **Плюсы:** Отлично подходят для задач, где мало данных на класс (one-shot/few-shot learning). Учат функцию сходства, а не классификацию.
- **Минусы:** Медленный инференс (нужно сравнивать с эталоном).
- **Задачи: Face verification/recognition**, отслеживание объектов (tracking), проверка подписи, поиск дубликатов.

RAFT (Recurrent All-Pairs Field Transforms) (2020) - для Optical Flow

- **Основная идея:** Итеративный подход к оценке оптического потока. Использует 4D корреляционный объем всех пар пикселей и GRU (рекуррентный блок) для итеративного обновления потока.
- **Плюсы: SOTA качество на стандартных бенчмарках**, хорошая обобщающая способность, относительно простая архитектура.
- **Минусы:** Высокие требования к памяти для 4D объема.

- **Задачи:** Оптический поток (определение движения каждого пикселя между кадрами) — основа для видеоаналитики, автономного вождения, компрессии видео.

Итоговая расширенная шпаргалка:

- **Нужен мощный и современный backbone?**
  - Для CNN: **ConvNeXt** (лучший "чистый" CNN) или **EfficientNetV2**.
  - Для Transformer: **Swin Transformer** (универсальный) или **ViT** (если данных очень много).
- **Мобильные/ограниченные устройства?**
  - **MobileNetV3** (сбалансированный), **EfficientNet-Lite**.
- **Детекция в 2024?**
  - Для практики и продакшена: **YOLOv8/v10** (лучший all-in-one фреймворк).
  - Для максимальной точности: **Faster R-CNN** с **ConvNeXt** или **Swin Transformer** backbone.
  - Для исследований: **DETR-вариации** (Deformable DETR, DINO).
- **Сегментация?**
  - Instance: **Mask R-CNN** (проверенный) или **Mask2Former** (новый SOTA).
  - Semantic: **DeepLabv3+** (с ResNet/ConvNeXt) или **U-Net** (если данных мало).
  - Для деталей и поз: **HRNet**.
- **Генерация изображений?**
  - **Diffusion Models** (Stable Diffusion) — текущий SOTA.
  - **GANs** (StyleGAN) — для узких задач и полного контроля.
- **Сравнение лиц или подобие?**
  - **Siamese Network** с **ResNet** backbone, обученная с **Triplet Loss** или **Contrastive Loss**.

Рекомендация: Для соревнований на Kaggle **начинайте с ResNet50/EfficientNet для классификации и YOLOv8/U-Net для детекции/сегментации**. Сначала побеждайте простыми, но надежными методами, а затем экспериментируйте с более сложными архитектурами из этого списка.