Методи екстракції локальних ознак в задачах розпізнавання визначних об'єктів місцевості

Дипломна робота на здобуття ступеня бакалавра

Виконала: Лочман Я.В., КА-41
Науковий керівник: доц. Яковлева А.П.

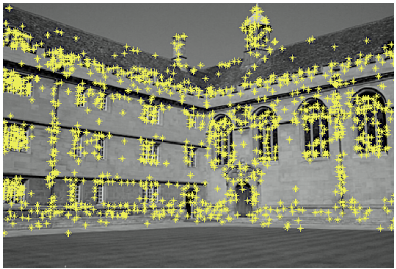
Національний Технічний Університут України "Київський Політехнічний Інститут ім. Ігоря Сікорського"
Навчально-Науковий Комплекс "Інститут Прикладного Системного Аналізу"
Кафедра Математичних Методів Системного Аналізу

Об’єкт дослідження

Алгоритми екстракції локальних ознак для розпізнавання об’єктів зображення.

Предмет дослідження

Застосування методів екстракції локальних ознак в задачі класифікації визначних об’єктів місцевості.



Мета

- дослідити теоретичні та практичні аспекти відомих методів екстракції локальних ознак зображень;
- провести аналіз та попередню обробку даних предметної області;
- проаналізувати та порівняти методи у поєднанні з алгоритмом класифікації – повнозв’язною нейронною мережею – в задачі розпізнавання визначних об’єктів місцевості.

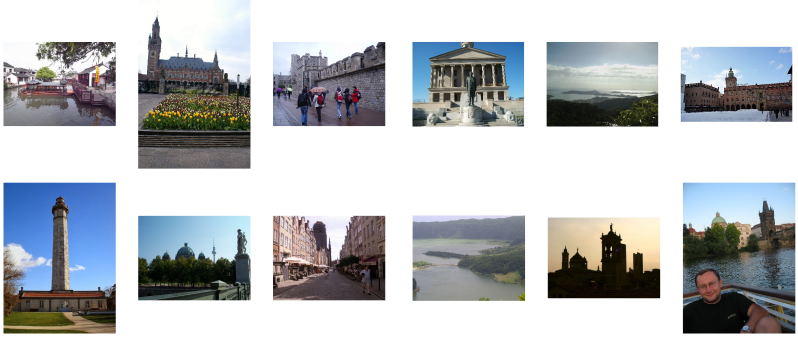
Локальні ознаки та їх дескриптори є фундаментом алгоритмів комп'ютерного зору. Розробка методів вирішення проблем розпізнавання інтенсивно розвивається, оскільки є потреба в ефективній обробці великої кількості цифрової інформації.

Спостерігається численне зростання колекцій фотографій з усіх точок Землі, і керування такими масивами унеможливлюється без належного тегування або групування зображень.

Задача розпізнавання (класифікації) зображень:

Вхідне зображення $\rightarrow$ клас або мітка з деякої фіксованої множини

Класифікуються визначні об'єкти місцевості – відомі місця світу:

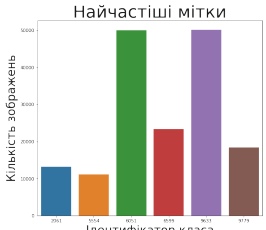


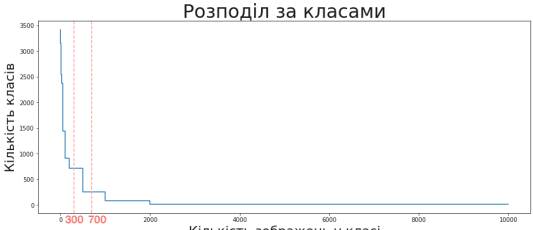
Використовується база даних – 1 000 000 фотографій з 15 000 точок світу.

Це – навчальна вибірка (для всіх зображень відомі класи – ідентифікатори локацій).

1. Нерепрезентативні класи + обмежені обчислювані потужності

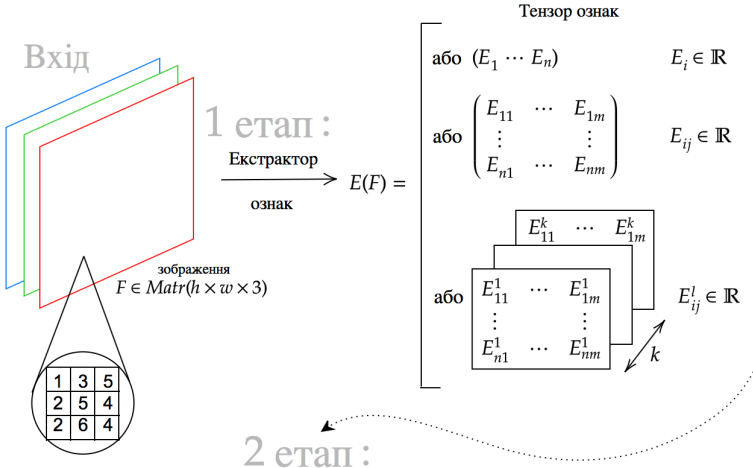
⇒ використання урізаної вибірки: 20 000 фотографій з 30ти класів (300-700 зобр. в кожному)





2. Приклад фотографій з одного класу:

⇒ вимагає від метода інваріантності відносно зсувів, поворотів, точок і кутів зйомки, масштабів, освітлення та передачі кольорів


Вхід
1 етап :
Екстрактор
ознак
E(F) =
Тензор ознак
або (E_1 ⋯ E_n)
E_i ∈ ℝ
або
E_ij ∈ ℝ
або
E^l_ij ∈ ℝ
k
зображення
F ∈ Matr(h × w × 3)
1 3 5
2 5 4
2 6 4
2 етап :
E(F)
Класифікатор
C⃗ = C(F) ∈ ℝ^c
Прогноз класу
(вектор впевненостей)
Вихід

Статичні методи. Характеризуються тим, що функція перетворення зображення будується за фіксованим алгоритмом, і всі параметри, від яких вона залежить (конфігурація фільтра, множина напрямків градієнта, значення порогу при фільтрації тощо) обираються емпіричним способом і задаються явно. Досліджуються:

- Алгоритм детектування країв Кенні (Canny edge detector)
- Гістограми напрямків градієнтів (HOG, Histograms of Oriented Gradients)
- Масштабонезалежне перетворення ознак (SIFT, Scale-Invariate Feature Transform)

Методи машинного навчання. На відміну від попередніх, є динамічними, тобто функція будується на основі представлених даних. В області глибокого навчання функцією є нейронна мережа, вона є композицією великої кількості простих функцій. Параметри, від яких вона залежить, які називають ваговими коефіцієнтами, знаходять ітеративно за допомогою оптимізації функції помилки. Розглядаються згорткові нейронні мережі:

- *CNN*1 – заснована на Resnet-50
- *CNN*2 – заснована на Inception v3
- *CNN*3 – заснована на Inception-ResNet v2

1. Згортка ч/б зображення з Гауссівським фільтром для згладжування

$$F_s = F * g_\sigma \qquad g_\sigma(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

2. Обрахування градієнта в кожній точці зображення

$$dF_x = F_s * \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \qquad dF_y = F_s * \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$G = \|\nabla F_s\| = \sqrt{dF_x^2 + dF_y^2}, \quad \theta = \tan^{-1}(dF_x / dF_y)$$

3. Пошук локальних максимумів

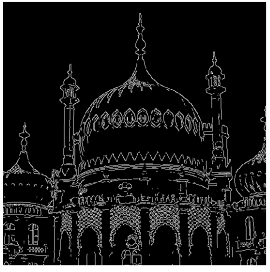
θ округлюється до ближчих $\tilde{\theta} \in \Theta = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$

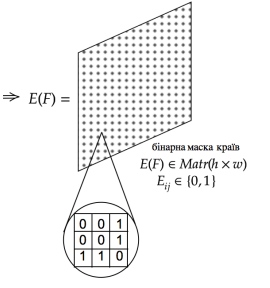
4. Гістерезисна фільтрація локальних максимумів

$$G(x,y) \geq t_{max} \Rightarrow \checkmark$$

$t_{min} < G(x,y) < t_{max}$, якщо в ε-околі $\exists(x_0, y_0) : G(x_0, y_0) \geq t_{max} \Rightarrow \checkmark$

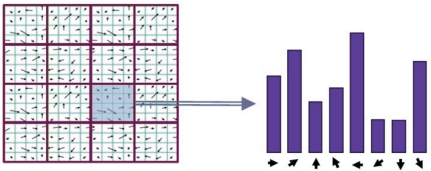
5. Формування карти ознак як матриці розміру вхідного зображення, кожна точка якої дорівнює або 1 (точка максимуму) або 0 (інакше)





1. Згладжування зображення (аналогічно)
2. Обрахування градієнтів (аналогічно)
3. Обрахування гістограм напрямків у клітинах $p_h \times p_w$ пікселів:

θ округлюються до ближчих $\tilde{\theta} \in \Theta = \{0, \frac{2\pi}{9}, \frac{4\pi}{9} \dots \frac{16\pi}{9}\}$

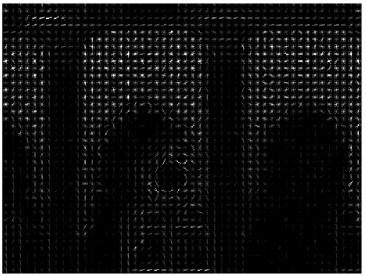


4. Об'єднання клітин в блоки $c_h \times c_w$, конкатенація гістограм, нормалізація:

$$\text{“L2-norm”}: \quad \nu_i \to \frac{\nu_i}{\sqrt{\|\nu_i\|_2^2 + \epsilon^2}}$$

5. Формування єдиного вектора ознак за рахунок конкатенації векторів гістограм кожного блоку в один

Візуалізація гістограм у блоках:



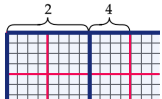
$E(F) \in \mathbb{R}^n$ – вектор гістограм напрямків

$$n = \left(\left(\left\lfloor \frac{h}{p_h} \right\rfloor - c_h + 1\right) \cdot c_h \cdot \left(\left\lfloor \frac{w}{p_w} \right\rfloor - c_w + 1\right) \cdot c_w\right) \cdot n_o$$

(p_h, p_w) – кількість пікселів у клітинці

(c_h, c_w) – кількість клітинок у блоці

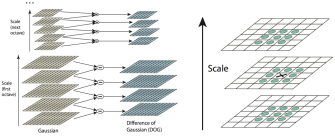
n_o – кількість проміжків напрямків



1. Побудова масштабного простору різниць гаусіанів

$$L(x, y, \sigma) = g(x, y, \sigma) * F(x, y) \qquad D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma)$$

2. Визначення множини локальних екстремумів



3. Уточнення локалізації ключових точок, фільтрація

$$D(\hat{\mathbf{x}}) \approx D(\mathbf{x}) + \left(\nabla D^T(\mathbf{x}), \hat{\mathbf{x}} - \mathbf{x}\right) + \frac{1}{2}\left(H(\mathbf{x})(\hat{\mathbf{x}} - \mathbf{x}), \hat{\mathbf{x}} - \mathbf{x}\right),$$

4. Визначення домінуючих напрямків градієнтів ключових точок

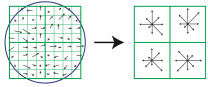
arg max гістограми напрямків градієнтів в точці або ≥ 0.8 max

5. Формування дескриптора ключових точок на основі гістограм напрямків градієнтів в околі точки

$$\Rightarrow \quad E(F) \in Matr(n \times m) \qquad E_{ij} \in R$$

матриця дескрипторів ключових точок

n – кількість знайдених ключових точок
$m = 128$ – розмірність вектора дескриптора ключової точки



Композиція простих функцій (шарів). Параметри, від яких вона залежить – вагові коефіцієнти – знаходять ітеративно за допомогою оптимізації функції помилки.

Розмірності вхідних векторів від екстракторів:

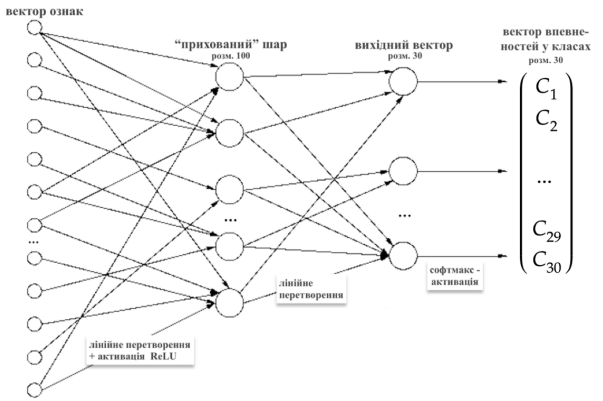
$$Canny: \quad 224 \times 224 \Rightarrow 112 \times 112 \Rightarrow 12544$$
$$HOG: \quad \left(\frac{224}{16} - 1 + 1\right)^2 * 8 = 1568$$
$$SIFT: \quad 128 \Rightarrow k - means_{100} \Rightarrow Tf - Idf \Rightarrow 100$$
$$SIFT_2: \quad 128 \Rightarrow k - means_{500} \Rightarrow Tf - Idf \Rightarrow 500$$

Функція двошарової нейронної мережі:

$$f(x) = a_2(\theta_2 \cdot a_1(\theta_1 \cdot x + b_1) + b_2)$$

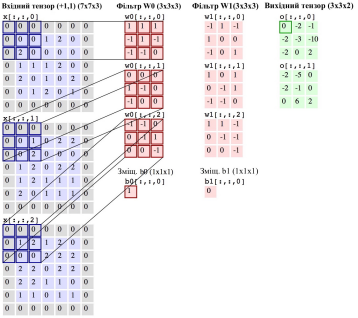
$a_1(x) = \max(0, x)$ – активація лін. випрямляч
$a_2(x) = \frac{e^x}{\sum_i e^{x_i}}$ – софтмакс-активація

Схема двошарової нейронної мережі:

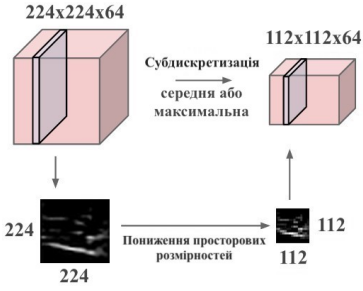


Підклас нейронних мереж, основні блоки – згортка та субдискретизація, що використовують просторову інформацію зображення. Усі складові функції:

Згортка



Субдискретизація



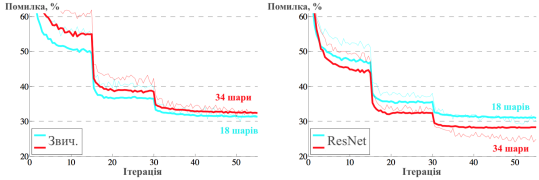
Активація лінійний випрямляч

$$a(x) = \max(0, x)$$

Софтмакс-активація

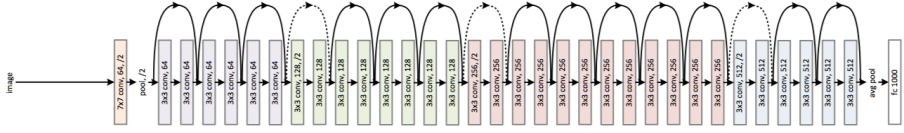
$$a(x) = \frac{e^x}{\sum_i e^{x_i}}$$

Порівняння графіків помилок алгоритму без та із залишковим блоком



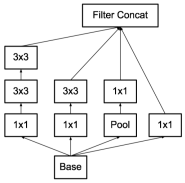
на еталонній вибірці (ImageNet)

Архітектура мережі ResNet:

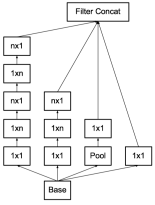


* Схема запозичена: He K. et al. Deep residual learning for image recognition.

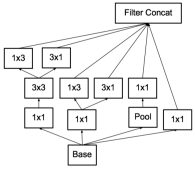
Блок Inception A



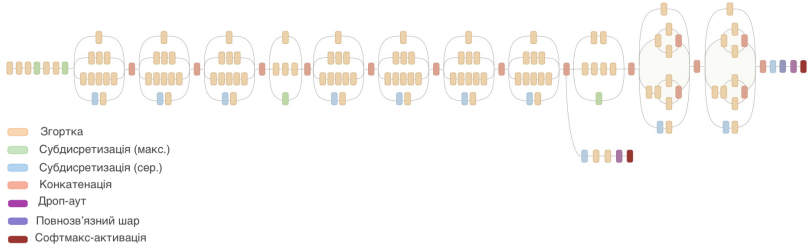
Блок Inception B



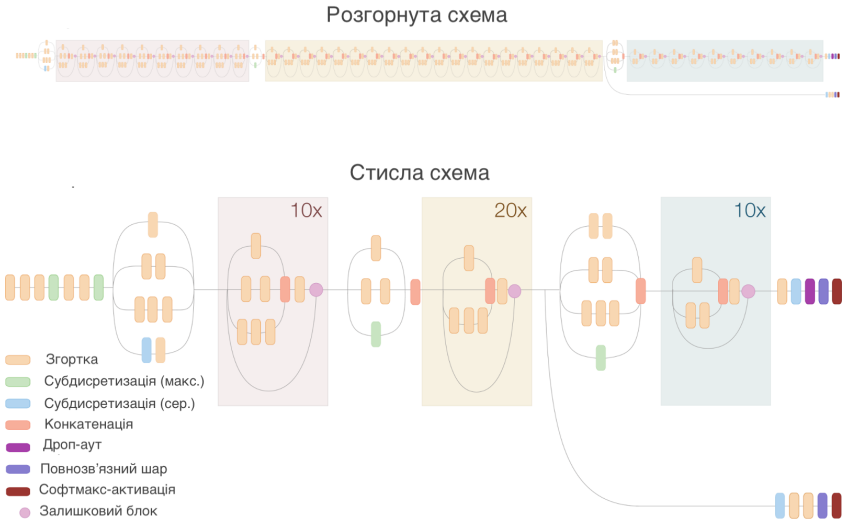
Блок Inception C



Архітектура мережі Inception:



* Схема запозичена: C. Szegedy et al. Rethinking the inception architecture for computer vision.



* Схема запозичена: C. Szegedy et al. Inception-v4, inception-resnet and the impact of residual connections on learning.

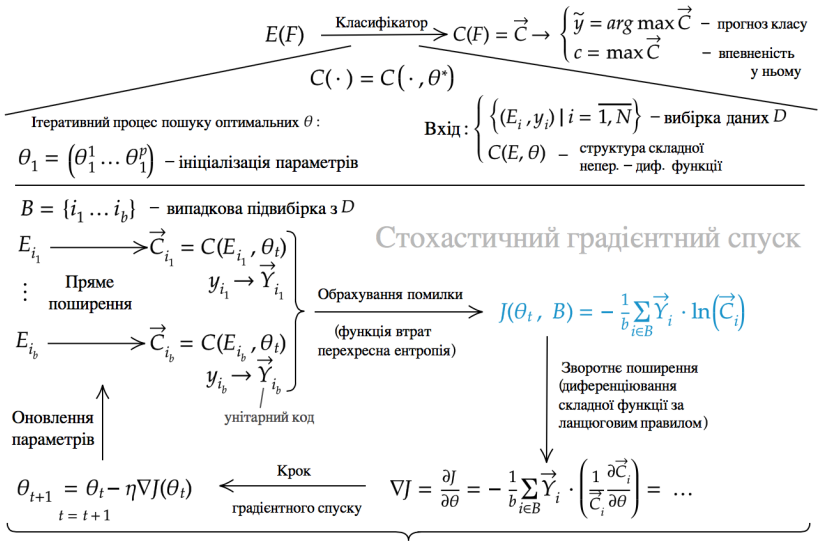
Попередня обробка: глобальна нормалізація контрастності

$$\mu = \frac{1}{N}\sum_{i=1}^{N} F_i, \quad \sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(F_i - \mu)^2}$$

$$\hat{F}_i = \frac{F_i - \mu}{\sigma}$$

Аугментації (поповнення набору даних за рахунок випадкових лінійних та інших перетворень)

- горизонтальні відзеркалювання
- зсуви в межах 10%
- повороти на макс. 15°
- стиснення/розтягування на макс. 20%
- випадкове кадрування до мін. 80%
- зміни кольору:
  - відхилення яскравості: $[-0.4, 0.4]$
  - відхилення контрасту: $[-0.4, 0.4]$
  - відхилення насиченості: $[-0.4, 0.4]$
  - відхилення відтінку: $[-0.07, 0.07]$
- фінальне кадрування: 224×224


$E(F)$
Класифікатор
$C(F) = \vec{C} \rightarrow \begin{cases} \tilde{y} = arg\max \vec{C} & \text{– прогноз класу} \\ c = \max \vec{C} & \text{– впевненість у ньому} \end{cases}$
$C(\cdot) = C(\cdot, \theta^*)$
Ітеративний процес пошуку оптимальних θ :
$\theta_1 = \left(\theta_1^1 \dots \theta_1^p\right)$ – ініціалізація параметрів
Вхід: $\begin{cases} \left\{(E_i, y_i) \mid i = \overline{1, N}\right\} & \text{– вибірка даних } D \\ C(E, \theta) & \text{– структура складної непер. – диф. функції} \end{cases}$
$B = \{i_1 \dots i_b\}$ – випадкова підвибірка з D
Стохастичний градієнтний спуск
$E_{i_1} \rightarrow \vec{C}_{i_1} = C(E_{i_1}, \theta_t)$
$y_{i_1} \rightarrow \vec{Y}_{i_1}$
Пряме поширення
$E_{i_b} \rightarrow \vec{C}_{i_b} = C(E_{i_b}, \theta_t)$
$y_{i_b} \rightarrow \vec{Y}_{i_b}$
унітарний код
Обрахування помилки
(функція втрат перехресна ентропія)
$J(\theta_t, B) = -\frac{1}{b}\sum_{i \in B} \vec{Y}_i \cdot \ln\left(\vec{C}_i\right)$
Зворотнє поширення (диференціювання складної функції за ланцюговим правилом)
Оновлення параметрів
$\theta_{t+1} = \theta_t - \eta \nabla J(\theta_t)$
$t = t + 1$
Крок градієнтного спуску
$\nabla J = \frac{\partial J}{\partial \theta} = -\frac{1}{b}\sum_{i \in B} \vec{Y}_i \cdot \left(\frac{1}{\vec{C}_i} \frac{\partial \vec{C}_i}{\partial \theta}\right) = \dots$
Цикл зупиняється, якщо $|J(\theta_t, B_1) - J(\theta_{t+1}, B_2)| < \varepsilon$ або після n ітерацій

Для валідаційного (тестового) набору даних $\{(F_i, y_i) \mid i \in T\}$ введемо критерії:

Доля правильних відповідей: $Acc_T = \left[\frac{TP + TN}{TP + TN + FP + FN}\right]_T$ (Accuracy)

Усереднена за класами середня точність: $MAP_T = \frac{1}{n_c}\sum_{c=1}^{n_c}\left[\int_0^1 p(r)dr\right]_T^{(c)}$ (Mean Average Precision)

Глобальна середня точність: $GAP_T = \frac{1}{|T|}\sum_{k=1}^{|T|} Acc_{\{i_1 \dots i_k\}} \cdot j_{\{\tilde{y}_{i_k} = y_{i_k}\}}$ (Global Average Precision)

де прогнози впорядковано за впевненістю: $c_{i_k} \geq c_{i_{k+1}} \forall k = \overline{1, |T|}$

Перехресна ентропія: $J_T = -\frac{1}{|T|}\sum_{i \in T} Y_i \ln(\vec{C}_i)$ (Cross-entropy)

використовувались допоміжні метрики (для множини індексів L):

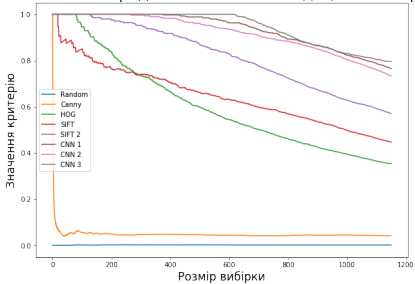
$TP_L = \sum_{i \in L} j_{\{\tilde{y}_i == y_i == 1\}}$ $\quad FP_L = \sum_{i \in L} j_{\{1 == \tilde{y}_i \neq y_i == 0\}}$ $\quad p_L = \frac{TP_L}{TP_L + FP_L}$ – **точність**

$TN_L = \sum_{i \in L} j_{\{\tilde{y}_i == y_i == 0\}}$ $\quad FN_L = \sum_{i \in L} j_{\{0 == \tilde{y}_i \neq y_i == 1\}}$ $\quad r_L = \frac{TP_L}{TP_L + FN_L}$ – **повнота**

Алгоритм	*Acc*	*MAP*	*GAP*	*J*	К-ть параметрів	Пам'ять
Вип. вгадування	3.30%	3.97%	0.19%	3.44	–	–
Canny + *MLP*	19.81%	9.16%	4.19%	3.26	1 257 530	4.8 Мб
HOG + *MLP*	45.96%	46.62%	35.38%	3.18	159 930	625 Кб
SIFT + *MLP*	56.56%	59.22%	48.70%	3.00	13 130	52 Кб
$SIFT_2$ + *MLP*	63.25%	68.02%	57.02%	2.88	53 130	208 Кб
*CNN*1	79.37%	76.17%	71.12%	2.24	23 596 062	91 Мб
*CNN*2	77.66%	75.96%	70.07%	2.14	21 829 822	84 Мб
*CNN*3	82.02%	80.21%	77.53%	2.01	54 322 302	208 Мб

- *Canny* + *MLP* – до зображення застосовується детектор Кенні, ескізи подаються на вхід класифікатора нейронної мережі
- *HOG* + *MLP* – для зображення обраховується вектор гістограм напрямків градієнтіві, після чого дескриптори ознак подаються на вхід класифікатора нейронної мережі.
- $SIFT_{(2)}$ + *MLP* – до зображення застосовується масштабонезалежне перетворення ознак, які подаються на вхід класифікатора нейронної мережі
- *CNN*1 – згорткова нейронна мережа, заснована на ResNet-50
- *CNN*2 – згорткова нейронна мережа, заснована на Inception v3
- *CNN*3 – згорткова нейромережа, засн. на Inception-ResNet v2

Глобальна середня точність на валідаційній вибірці



- Усі три моделі CNN перевершують засновані на SIFT в задачі розпізнавання об’єктів.
- В свою чергу SIFT випереджає алгоритми, засновані на Canny та HOG. Найгірші результати дав метод із екстракцією країв Кенні, причиною може бути те, що виявлених за цим алгоритмом ознак недостатньо для правильної класифікації.
- Алгоритм SIFT дає досить високі результати при відносно незначних використаннях обчислюваних потужностей – час роботи алгоритма у декілька разів менший за час навчання згорткових нейронних мереж; і кількість параметрів, за якими оптимізується класифікатор, на порядки менше за моделей CNN.
- У підсумку пропонується використовувати згорткову нейронну мережу, засновану на архітектурі ResNet за наявності потужних обчислюваних ресурсів, оскільки вона є золотою серединою між якістю роботи та затратами пам’яті / часу
- Також пропонується використовувати масштабонезалежне перетворення ознак разом із повонозв’язною нейронною мережею в якості класифікатора в разі, якщо обчислюваних потужностей недостатньо.

- Використання більш потужних графічних процесорів для тренування моделей (хмарні сервіси: Amazon Web Services, Microsoft Azure, Google Cloud тощо) $\Rightarrow$ можливість порівняти методи на всій вибірці
- Згорткова нейронна мережа Xception (2017) (оптимізована архітектура моделі Inception – 20 868 422 параметрів)
- Згорткова нейронна мережа DenseNet (2018) (за останніми показниками на еталонних вибірках даних має високі результати)

Дякую за увагу!