УДК 004.93

А.М. Літвинчук, Л.В. Барановська

ПОКРАЩЕННЯ МОДЕЛЕЙ РОЗПІЗНАВАННЯ ОБЛИЧ ЗА ДОПОМОГОЮ НАВЧАННЯ ПОДІБНОСТІ, РОЗКЛАДУ ЗМІНИ ТЕМПУ НАВЧАННЯ ТА АУГМЕНТАЦІЙ

Ключові слова: згорткові нейронні мережі, розпізнавання облич, навчання подібності, розклад зміни темпу навчання, аугментації.

Ключевые слова: сверточные нейронные сети, распознавания лиц, обучение подобности, расписание смены темпа обучения, аугментации.

Вступ

Розпізнавання облич — це одна з основних задач комп'ютерного зору, актуальна в силу її практичної значущості, що викликає велику зацікавленість широкого кола науковців. І хоча дослідження у сфері відбувались з початку розвитку комп'ютерного зору, адекватних результатів змогли досягнути лише за допомогою згорткових нейронних мереж.

Мета роботи — дослідження різних методів розпізнавання облич за допомогою навчання подібності, зміни розкладу темпу навчання та аугментацій, а також їх комбінування задля визначення якісного і надійного алгоритму навчання нейронних мереж для задачі розпізнавання облич.

Сфера розпізнавання облич до нейронних мереж

Розглянемо сферу розпізнавання облич до нейронних мереж, а також покажемо мінуси кожного з підходів.

1. Метод головних компонент. Це один з найвідоміших та найбільш якісно опрацьованих методів розпізнавання [1]. Він широко відомий у стандартному машинному навчанні, зокрема використовується для пониження розмірності вхідних векторів ознак для більш ефективного та точного вирішення задачі.

Зменшення вектора розмірності пояснюється тим, що ми хочемо з обличчя отримати вектор «базових» ознак, які можуть бути спільними у різних людей. Цей підхід є основою для багатьох методів, зокрема для згорткових нейронних мереж, проте спосіб отримання цього вектора базових ознак у кожного методу різний. Метод головних компонент аналізує набір тренувальних даних та виявляє зміну вхідних векторів ознак, після чого описує цю зміну в базисі власних векторів. Таким чином власне число при власному векторі буде показувати його важливість.

Хоча метод добре математично обгрунтований та досліджений, на жаль, його результати у реальних задачах досить погані. Його плюсом є лише швидкість роботи: множення вектора на матрицю — дуже швидка операція, в результаті даний метод може обробляти тисячі фотографій в секунду. Проте недостатньо місткий для хорошого розпізнавання великої кількості облич, не дивлячись на те, що покращується з великою кількістю вхідних даних. Також метод не стійкий до різних деформацій обличчя та шуму, часто наявному у фотографіях.

2. Активні моделі зовнішнього вигляду. Дані моделі зовнішнього вигляду, як і метод головних компонент — статистичні моделі, які за рахунок деформацій підганяються під реальне зображення.

Ця модель включає в себе два типи параметрів: параметри, пов'язані з формою, а також параметри, пов'язані зі статистичною моделлю пікселів зображення та текстурою. Щоб навчити цю модель, потрібна повністю ручна розмітка даних. Кожне обличчя розбиваємо майже на 70 характерник точок, які модель буде вчитись адаптувати до нового зображення.

За допомогою активних моделей зовнішнього вигляду можна моделювати фотографії об'єктів із різними деформаціями. Форма лиця моделюється частиною параметрів, а іншою частиною моделюється текстура. Деформація у цьому випадку — масштабування, перенесення, поворот [2]. Цей метод можна використовувати як для детекції обличчя, так і для розпізнавання [3].

З явних плюсів можна ще виділити швидкість роботи, оскільки модель має не дуже багато параметрів. Також вона більш стійка до трансформацій, ніж метод головних компонет, проте, як і метод головних компонент, не є місткою, а значить, і точною, незважаючи на статистичність [3].

Опис підходів

Розглянемо архітектуру нейронної мережі, підхід для навчання подібності, різні розклади темпу навчання та аугментації.

1. Архітектура нейронної мережі. У цій роботі використовували SE-ResNet50 як єдину мережу для експериментів, оскільки показано [3], що архітектура достатньо містка для даної задачі.

2. Навчання подібності (підхід ArcFace). Незважаючи на простоту, цей метод надзвичайно ефективний у задачі розпізнавання облич, тому і дістав відповідну назву. Він пропонує новий спосіб підрахунку ймовірностей класів під час тренувального процесу [4]. Попередньо цей метод детально аналізували у роботі [3].

Нехай у нас є зображення, яке належить класу J, а всього класів N, нейронна мережа формує базовий вектор $x \in R^d$. Тоді останній повнозв'язний шар матиме лише ваги W з розмірністю (d, N), зсуву у нього немає. Розглянемо наступні величини [3]:

$$x^{norm} = \frac{x}{\|x\|_2},$$

$$W_j^{norm} = \frac{W_j}{\| W_j \|_2} \quad \forall j = 1, \ldots, N,$$

$$Logits = x^{norm} W^{norm},$$

$$Logits_J = \mathrm{Cos}(x^{norm}, W_J^{norm}) = \cos(\theta_J),$$

$$\theta_j = \arccos(\mathrm{Cos}(x^{norm}, W_j^{norm})) = \arccos(Logits_j) \; \forall j = 1, \ldots, N,$$

$$\begin{cases} Logits_j = \cos(\theta_j) & \forall j \neq J, \\ Logits_J = \cos(\theta_J + m), \end{cases}$$

$$Logits_j = s \cdot Logits_j \; \forall j = 1, \ldots, N.$$

3. Метод оптимізації нейронних мереж. Для оптимізації використано стохастичний градієнтий спуск з моментом, оскільки раніше було показано, що він є досить хорошим для задачі розпізнавання облич [3]. Зміна параметрів у цьому методі відбувається за таким правилом:

$$h_t = \beta \cdot h_{t-1} + (1-\beta) \cdot \nabla_\theta \sum_{i=1}^{M} L(f(x^{(i)}, \theta), y^{(i)}),$$

$$\theta_{t+1} = \theta_t - \alpha_t \cdot h_t.$$

Тут α_t — темп навчання, L — функція втрат, f — наша нейронна мережа.

4. Розклад зміни темпу навчання. Параметр α_t надзвичайно важливий. Якщо його поставити занадто великим, то мережа просто не буде вчитись, а якщо занадто малим — то може дуже довго збігатись.

Контролювати процес навчання можна за допомогою розкладу зміни темпу навчання, тобто зміна темпу навчання, як функції від часу. Розглянемо різні розклади, їх переваги та недоліки.

Статичний темп навчання має такий вигляд: $\alpha_t = \alpha \; \forall t,$ де α — певне значення, яке оберемо до навчання.

Багатокроковий розклад темпу навчання запишемо

$$\alpha = \begin{cases} \alpha, & 0 \le t \le T_1, \\ \alpha \cdot \gamma, & T_1 \le t \le T_2, \\ \alpha \cdot \gamma^2, & T_2 \le t \le T_3, \\ \dots & \\ \alpha \cdot \gamma^{N-1}, & T_{N-1} \le t \le T_N, \end{cases}$$

де $T_1, ..., T_N$ — певні рубежі; γ — певне число, на яке будемо домножувати темп навчання після перетинання рубежу.

Як тільки кількість ітерацій перетнула певний рубіж, відразу множимо попередній темп навчання на γ. Цей розклад навчання дозволив набагато краще навчати нейронні мережі, проте у ньому багато параметрів, які потрібно підбирати під задачу: α, кількість рубежів, їх значення, γ.

Для наступного методу, який дістав назву зменшення темпу навчання на плато, обов'язково потрібна валідаційна вибірка, на якій вимірюватимемо точність (або іншу цільову метрику). Алгоритм зменшення виглядає наступним чином: якщо цільова метрика не покращується протягом T ітерацій, то множимо попередній темп навчання на γ:

$$\alpha_t = \alpha_{t-1} \cdot \gamma.$$

Параметр T називається терпінням розкладу. Цей розклад практично ідентичний попередньому, проте рубежі ставляться не як константи, а є динамічними і залежать від якості на валідаційній вибірці.

Косинусний розклад [5]:

$$\alpha_t = \alpha_{\min} + \frac{1}{2}(\alpha_{\max} - \alpha_{\min})\left(1 + \cos\left(\frac{t}{T}\right)\right),$$

де $\alpha_{\min}$ — мінімальний допустимий темп навчання (зазвичай ставлять дуже мале значення); T — кількість ітерацій у навчанні.

Оскільки тут взагалі немає ніяких рубежів і параметра γ, цей розклад дуже зручний у використанні. Також він поступово зменшує темп навчання згідно нелінійної функції і в результаті робить темп дуже повільним, що збільшує якість знайдених локальних мінімумів. Косинусний розклад вважається найкращим розкладом темпу навчання.

Подальші методи є покращеннями попередніх підходів. Комбінуючи їх, можна скласти розклад, який приводитиме до кращих результатів.

Розминка темпу навчання практично необхідна завжди, зокрема, коли розмір міні-партії дуже великий. Також вона покращує результат роботи адаптивних методів ґрадієнтного спуску. Розглянемо лінійну розминку:

$$\alpha_t = \frac{t}{T_{warmup}} \cdot \alpha,$$

де T_{warmup} — період розминки; α — початковий темп навчання, тобто поступово, починаючи від нуля, збільшуємо темп навчання, поки він не буде рівний темпу, з якого починали навчання. Після цього можна починати будь-який інший розклад.

Косинусний розклад дуже часто використовують з оновленнями. Оновлення зводять відновлення темпу навчання до початкового. Таким чином, навчання ділиться на періоди: в кожному періоді змінюємо темп по косинусному розкладу, проте в кінці періоду змінюємо темп на початковий і починаємо новий період. Зазвичай кожен період збільшує свою довжину. Тоді модифікований розклад буде наступним:

$$\alpha_t = \alpha_{\min} + \frac{1}{2}(\alpha_{\max} - \alpha_{\min})\left(1 + \cos\left(\frac{T_{cur}}{T_i}\right)\right),$$

$$T_i = T_{mult} \cdot T_{i-1},$$

де T_{cur} — кількість ітерацій, що мали місце після початку періоду; T — кількість ітерацій в i-му періоді; T_{mult} — кількість ітерацій у наступному періоді залежно від попереднього.

На рис. 1 зображено косинусний графік зміни темпу навчання з оновленнями [6] та параметрами $T_0 = 100, \alpha_{\max} = 0{,}1, \alpha_{\min} = 0, T_{mult} = 1.$

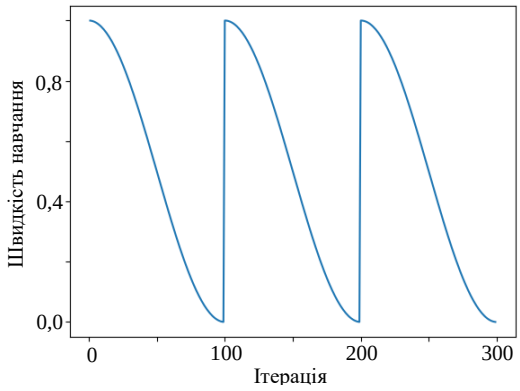

Рис. 1

Також використаємо так зване заспокоєння. Метод полягає в тому, щоб після закінчення косинусного розкладу ще деякий час продовжувати навчання з мінімальним темпом навчання $\alpha_{\min}$. Це аргументується тим, що у цьому випадку можемо обійти локальний мінімумум, у якому опинилась нейронна мережа, та знайти кращі параметри.

Ще один приклад комбінування різних методів змін темпу навчання знаходиться на рис. 2. Тут комбінуємо розминку з $T_{warmup} = 5,$ косинусний розклад без оновлень з $\alpha_{\max} = 0{,}5,$ $\alpha_{\min} = 0{,}01, T = 15$ та $T_{cooldown} = 10$ (заспокоєння) [7].

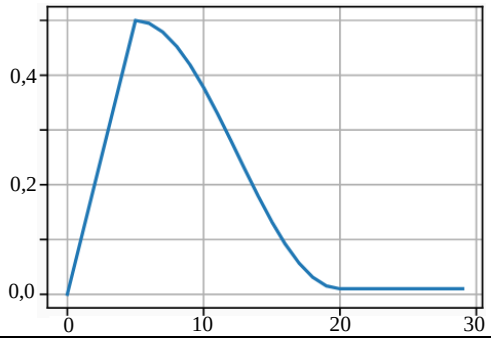

Рис. 2

У даній роботі використовуємо різні розклади та порівнюватимемо їх для задачі розпізнавання облич.

5. Аугментації. Перенавчання є серйозною проблемою для нейронних мереж, зокрема тому, що мають дуже багато параметрів, а даних зазвичай не настільки багато, щоб гарантувати узагальнюючу можливість моделі. Додаткова розмітка даних може займати багато часу і бути доволі дорогою, тому виник такий підхід, як аугментації. Розглянемо означення аугментації для класифікації зображень.

Аугментація — це процес трансформування зображення і додавання його до тренувального набору даних, при цьому клас зображення не змінюється. Таким чином мережа побачить одну й ту саму картинку у різних формах, з шумом та без, з різними поворотами, зміною кольорів, котрастності та яскравості. Фактично, це метод регуляризації — фіктивно збільшуємо кількість навчальних даних різними трансформаціями картинок з тренувальної вибірки, при цьому точно знаємо, що клас у них при цих трансформаціях не змінився. Мережа має бути стійкою до різних шумів, до місця знаходження певних об’єктів на зображенні та до різних кольорових зміщень. Для аугментацій будемо використовувати бібліотеку albumentations [8].

Розглянемо деякі приклади аугментацій для зображення на рис. 3.

На рис. 4 можна побачити збільшення контрасту та яскравості.

Рис. 3

Рис. 4

На рис. 5 зображено зміну кольорової гами.

На рис. 6 можна побачити збільшене та повернуте зображення.

Рис. 5

Рис. 6

Незважаючи на всі трансформації, клас зображення не змінився — це папуга. Аугментації можна комбінувати, щоб отримати нові, унікальні зображення. Проте не будь-які трасформації покращують якість моделі. Наприклад, на рис. 7 зображена аугментація, яка погано вплине на якість моделі.

Рис. 7

Вертикальний поворот зазвичай погано себе показує у задачах класифікації, зокрема у задачі розпізнавання облич. Він руйнує певні уявлення мережі про клас, наприклад, після такої трансформації мережа може завчити, що клюв може бути нахиленим вверх — хоча фактично у реальному світі така ситуація не зустрічається.

Результати

Для порівняння підходів взяли VGGFace2 для навчання та валідації, Accuracy та Accuracy@5 як основні метрики та наступну попередню обробку даних [3]:

1) вирівнення обличчя;

2) зміна розміру зображення до (112, 112, 3);

3) нормалізація.

Для всіх експериментів використовували розмір міні-партії 540, 80 епох та початковий темп навчання 0,1.

Щоб зробити певний експеримент, від якого будемо відштовхуватись, використовуватимемо наступні параметри мережі: архітектура: SEResNet-50; без підходу навчання подібності, розклад. Результати: Accuracy = 0,714, Accuracy@5 = = 0,74.

Тепер візьмемо попередній експеримент і просто додамо навчання подібності, а саме ArcFace з параметрами $s = 24$ та $m = 0{,}2$, залишивши усі інші параметри сталими. Результати: Accuracy = 0,88, Accuracy@5 = 0,94. Бачимо, що просто додавши концепцію навчання подібності, ми отримали великий приріст, а саме 17 % точності.

Результати цього блоку експериментів показані у табл. 1.

Таблиця 1

Arcface $s=32$, $m=0{,}2$	Accuracy	Accuracy@5
–	0,714	0,74
+	0,88	0,94

Після цього поекспериментуємо з аугментаціями. Під легкими аугментаціями маємо на увазі зміну яскравості на контрастності та зміну кольорової гами. Більш важкі аугментації включають легкі аугментації з більшою ймовірністю застосування та більш радикальними параметрами, а також аугментації стискання та додавання нормального шуму до зображення. Аугментація повороту і збільшення обличчя доволі специфічна, тому її винесли окремо. Як бачимо з табл. 2, у цій задачі важливо використовувати важкі аугментації

з аугментацією повороту і збільшення. Також варто підмітити, що аугментація повороту і збільшення завжди збільшує точність, на нашу думку, це тому, що вирізання і вирівнювання обличчя не завжди відбувається ідеально, тому корисно дати мережі на навчання трохи повернуті і збільшені обличчя для імітації цих артефактів.

Таблиця 2

Легкі аугментації	Важкі аугментації	Аугментація повороту і збільшення	Accuracy	Accuracy@5
+	–	–	0,885	0,941
+	–	+	0,891	0,945
–	+	–	0,902	0,948
–	+	+	0,906	0,95

Завершимо наші експерименти підбором розкладу зміни темпу навчання. Результати останнього блоку експериментів знаходяться у табл. 3.

Таблиця 3

Багатокроковий розклад	Зменшення темпу навчання на плато	Косинусний розклад	Косинусний розклад з розігрівом та заспокоєнням	Accuracy	Accuracy@5
+	–	–	–	0,906	0,95
–	+	–	–	0,912	0,954
–	–	+	–	0,931	0,96
–	–	–	+	0,935	0,961

Отже, виходячи з наших експериментів, найкраща конфігурація для навчання моделі розпізнавання облич для цього навчального набору даних наступна:

1) ArcFace з параметрами $s = 24$ та $m = 0{,}2$;

2) важкі аугментації з поворотом та збільшенням обличчя.

Висновок

У даній роботі перевірено ряд гіпотез щодо покращення якості роботи мережі для розпізнавання облич. У більшості випадків підходи, які в теорії мали давати кращі результати, дійсно це робили.

Зокрема, навчання подібності критичне для задачі розпізнавання облич. Без цього підходу задача невирішувана. Важливо використовувати аугментації як спосіб регуляризації моделі. Також варто використовувати косинусний розклад, оскільки він зможе покращити результати моделі навіть тоді, коли здається, що мінімум уже досягнуто.

Загалом, використовуючи всі перелічені підходи, ми змогли отримати точність 93,5 % на досить складному наборі даних, що на 22 % краще за базовий експеримент.

А.М. Літвинчук, Л.В. Барановська

ПОКРАЩЕННЯ МОДЕЛЕЙ РОЗПІЗНАВАННЯ ОБЛИЧ ЗА ДОПОМОГОЮ НАВЧАННЯ ПОДІБНОСТІ, РОЗКЛАДУ ЗМІНИ ТЕМПУ НАВЧАННЯ ТА АУГМЕНТАЦІЙ

Розпізнавання облич — одна з основних задач комп'ютерного зору, актуальна в силу її практичної значущості та викликає велику зацікавленість широкого кола науковців. І хоча дослідження у сфері відбувались з початку розвитку комп'ютерного зору, адекватних результатів змогли досягнути лише за допомогою згорткових нейронних мереж. У даній роботі проведено порівняльний аналіз методів розпізнавання обличчя до згорткових нейронних мереж. Розглянуто метод навчання подібності, аугментації та розклади зміни темпу навчання. Проведено ряд експериментів, виконано порівняльний аналіз розглянутих методів покращення згорткових нейронних мереж, у результаті отримано універсальний алгоритм для навчання моделі розпізнавання облич. У роботі використано SE-ResNet50 як єдину мережу для експериментів. Навчання подібності — це метод, за допомогою якого можливо досягнути достатньої точності. Перенавчання є серйозною проблемою для нейронних мереж, зокрема тому, що мають дуже багато параметрів, а даних зазвичай не настільки багато, щоб гарантувати узагальнюючу можливість моделі. Додаткова розмітка даних може займати багато часу і бути доволі дорогою, тому виник такий підхід, як аугментації. Аугментації штучно збільшують тренувальний набір даних, тому цілком природньо, що метод аугментації у всіх експериментах покращив результати відносно початкового експерименту. Різні степені та більш агресивні форми аугментації у задачі розпізнавання облич у даній роботі приводив до кращих результатів. Як і очікувалось, найкращим розкладом зміни темпу навчання виявився косинусний з розігрівом та оновленнями. Цей розклад має мало параметрів, до того ж зручний у використанні. Загалом, використовуючи різні підходи, отримали точність 93,5 % на досить складному наборі даних, що на 22 % краще за базовий експеримент. У наступних дослідженнях планується розглянути покращення не лише моделі розпізнавання облич, а й детекції. Від якості детекції обличчя безпосередньо залежить точність розпізнавання.

Ключові слова: згорткові нейронні мережі, розпізнавання облич, навчання подібності, розклад зміни темпу навчання, аугментації.

A.M. Litvynchuk, L.V. Baranovska

IMPROVING FACE RECOGNITION MODELS USING METRIC LEARNING, LEARNING RATE SCHEDULERS, AND AUGMENTATIONS

Face recognition is one of the main tasks of computer vision, which is relevant due to its practical significance and great interest of wide range of scientists. It has many applications, which has led to a huge amount of research in this area. And although research in the field has been going on since the beginning of the computer vision, good results could be achieved only with the help of convolutional neural networks. In this work, a comparative analysis of facial recognition methods before convolutional neural networks was performed. A metric learning approach, augmentations and learning rate schedulers are considered. There were performed bunch of experiments and comparative analysis of the considered methods of improvement of convolutional neural networks. As a result a universal algorithm for training the face recognition model was obtained. In this work, we used SE-ResNet50 as the only neural network for experiments. Metric learning is a method by which it is possible to

 ISSN 1028-0979

achieve good accuracy in face recognition. Overfitting is a big problem of neural networks, in particular because they have too many parameters and usually not enough data to guarantee the generalization of the model. Additional data labeling can be time-consuming and expensive, so there is such an approach as augmentation. Augmentations artificially increase the training dataset, so as expected, this method improved the results relative to the original experiment in all experiments. Different degrees and more aggressive forms of augmentation in this work led to better results. As expected, the best learning rate scheduler was cosine scheduler with warm-ups and restarts. This schedule has few parameters, so it is also easy to use. In general, using different approaches, we were able to obtain an accuracy of 93,5 %, which is 22 % better than the baseline experiment. In the following studies, it is planned to consider improving not only the model of facial recognition, but also detection. The accuracy of face detection directly depends on the quality of face recognition.

Keywords: convolutional neural networks, face recognition, metric learning, learning rate schedulers, augmentations.

1. Perlibakas V. Face recognition using principal component analysis and Log-Gabor filters. *Conference on Computer Vision and Pattern Recognition*. 2005. https://arxiv.org/abs/cs/0605025.
2. Zhao W., Chellapa R. Image-based face recognition: issues and methods. *Conference on Computer Vision and Pattern Recognition.* 2002. https://www.face-rec.org/interesting-papers/general/chapter_figure.pdf.
3. Літвинчук А.М., Барановська Л.В. Покращення моделей розпізнавання облич за допомогою згорткових нейронних мереж, навчання подібності та методів оптимізації. *Международный научно-технический журнал «Проблемы управления и информатики».* 2021. № 5. С. 140–149.
4. Deng J., Guo J., Zafeiriou S., Xue N. ArcFace: additive angular margin loss for deep face recognition. *Conference on Computer Vision and Pattern Recognition.* 2018. https://arxiv.org/abs/1801.07698.
5. Loshchilov I., Hutter F. SGDR: stochastic gradient descent with warm restarts. I*nternational Conference on Learning Representations.* 2016. https://arxiv.org/abs/1608.03983.
6. Pechyonkin M. Stochastic weight averaging — a new way to get state of the art results in deep learning. 2018. https://towardsdatascience.com/stochastic-weight-averaging-a-new-way-to-get-state-of-the-art-results-in-deep-learning-c639ccf36a.
7. Learning rate scheduling. https://d2l.ai/chapter_optimization/lr-scheduler.html.
8. Buslaev A., Parinov A., Kvedchenya E., Iglovikov V. Albumentations: fast and flexible image augmentations. *Conference on Computer Vision and Pattern Recognition.* 2018. https://arxiv.org/abs/1809.06839.

Отримано 25.08.2021