POLITECHNIKA ŚLĄSKA W GLIWICACH
WYDZIAŁ INŻYNIERII BIOMEDYCZNEJ

Sprawozdanie

Biocybernetyka
PROJEKT: Algorytm rozpoznawania twarzy z użyciem konwolucyjnych sieci neuronowych

Natalia Wyleżoł

Zabrze, 16 grudnia 2019

Spis treści

Spis rysunków

1. Wprowadzenie do tematu

1.1 Wstęp

Zdolność ludzkiego mózgu do rozpoznawania i zapamiętywania twarzy jest niezwykła zważywszy na to, że w ciągu swojego życia możemy zapamiętać setki, jak nie tysiące twarzy. Co więcej, jeśli osoba nie jest dotknięta prozopagnozją ani zaburzeniami widzenia (zaburzenie percepcji wzrokowej, polegające na upośledzeniu zdolności rozpoznawania twarzy znajomych lub widzianych już osób, a w niektórych przypadkach także ich wyrazu emocjonalnego), może ona rozpoznać twarze patrząc pod różnym kątem, w różnym oświetleniu, nawet po niewielkich zmianach w wyglądzie typu nowa fryzura czy okulary.

1.2 Historia

Naukowcy od dawna pracują nad systemem imitującym umiejętność mózgu do rozpoznawania twarzy. W 1966, Bledsoe, stworzył system, który potrafił sklasyfikować zdjęcia twarzy używając algorytmu łańcucha kodów (chain code). [1] Był też pierwszym, który próbował stworzyć półautomatyczny algorytm rozpoznawanie twarzy. Polegał on na ręcznym zaznaczaniu twarzy przez użytkownika, dzięki czemu komputer mógł zaklasyfikować ją do odpowiedniej osoby. [2]

W 1987 Sirovich i Kriby [3] wykazali, że na podstawie analizy cech twarzy można wyróżnić zestaw cech tych najbardziej podstawowych. Wykazali również, że potrzebnych jest mniej niż 100 wartości aby dokładnie opisać znormalizowaną twarz.

In 1991, Turk i Pentland [4] rozwinęli metodę Eigen face dzięki wynalezieniu sposobu detekcji twarzy w obrazie. Był to pierwszy krok do zautomatyzowania algorytmu rozpoznawania twarzy.

Od 1993 do 2000 the Defense Advanced Research Projects Agency (DARPA) i National Institute of Standards and Technology weszły na rynek z nową technologią rozpoznawania twarzy (FERET) [5], na którą składało się stworzenie bazy danych zawierającej zdjęcia twarzy. Znajduje się w niej 2413, 24-bitowych kolorowych zdjęć przedstawiających 856 ludzi.

W 2010 roku nastąpiła wielka zmiana dla mediów społecznościowych i ich użytkowników na całym świecie, a mianowicie zaczęto pracę nad tagowaniem zdjęć na których pojawiała się twarz. Jednakże dokładność algorytmu nie była wystarczająca, dlatego powstały technologie używające głębokiego uczenia (deep learning) takie jak deep face.[7] Były one złożone z dziewięciowarstwowej sieci neuronowej z ponad 120 milionami połączeń, których trening odbywał się z użyciem czterech milionów zdjęć przesłanych przez użytkowników Facebooka.

1.3 •

Komputerowe rozpoznawanie obrazu jest technologią łączącą przetwarzanie obrazów z tzw. widzeniem komputerowym. Pozwala ona na skuteczne rozpoznawanie na obrazie wybranego obiektu, takiego jak twarz, budynki, zwierzęta, samochody, charakter pisma itp. Mimo tego, że inne sposoby identyfikacji, takie jak odciski palców czy skanowanie tęczówki oka, mogą być bardziej precyzyjne, wciąż prowadzi się badania nad rozpoznawaniem twarzy, ponieważ jest to metoda nieinwazyjna. Większość badań skupia się na detekcji indywidualnych cech twarzy takich jak oczy, nos, usta, obrys twarzy i opracowywaniu modelu z uwzględnieniem umiejscowienia, rozmiaru i względnego położenia pomiędzy tymi cechami. Jednakże takie podejście jest trudne do opracowania. Badania nad działaniem ludzkiego rozpoznawania twarzy pokazało, że indywidualne cechy i ich bezpośrednie relacje nie są wystarczające, aby identyfikacja dorosłych ludzi była w pełni wydajna. [9] Nie mniej jednak, takie podejście pozostaje najpopularniejsze w literaturze komputerowego widzenia.

1.4 Zastosowania

Jak już wcześniej wspomniano, rozpoznawanie twarzy jak i sama jej detekcja może mieć różne zastosowania:

- Automatyczna identyfikacja - używana do identyfikacji osoby (karty kredytowe, prawo jazdy, paszport, identyfikator pracownika) w kontroli dostępu,
- Identyfikacja osób poszukiwanych lub zaginionych,
- Interakcja Człowiek-Komputer - sterowanie komputerem za pomocą ruchów głowy, np. w grach komputerowych (gracz musi założyć na głowę specjalny sprzęt),
- Monitoring - wykrywanie i śledzenie twarzy może służyć zwiększeniu bezpieczeństwa w miejscach publicznych, takich jak lotniska, centra handlowe, koncerty czy na prywatnych posesjach.
- Filtry w aplikacjach (np. Facebook, Snapchat),
- Autoryzacja, logowanie do konta, odblokowanie telefonu

2. Konwolucyjne sieci neuronowe

2.1 Podstawy

Konwolucyjne sieci neuronowe, znane też jako CNN czy ConvNet, są sztucznymi sieciami neuronowymi, które najczęściej mają zastosowanie w analizie obrazów. Oprócz tego, mogą być wykorzystywane do innej analizy danych jak i zagadnienia klasyfikacji. Ogólnie można myśleć o konwolucyjnej sieci neuronowej jako o zwykłej, sztucznej sieci neuronowej, która jest wyspecjalizowana w detekcji i rozpoznawaniu wzorów. Dzięki tej umiejętności konwolucyjne sieci neuronowe są niezwykle przydatne w przetwarzaniu obrazów czy sygnałów.

2.2 Architektura sieci

Każda konwolucyjna sieć neuronowa składa się z 3 podstawowych warstw: konwolucyjnej, warstwy poolingu i warstwę spłaszczającą.

2.2.1 Warstwa konwolucyjna

Neurony tej warstwy zachowują się podobnie do tradycyjnych - otrzymują dane wejściowe, przekształcają je i podają na wejście neuronu kolejnej warstwy. Dane zostają przekształcone za pomocą operacji splotu (ang. convolution - splot). Te konkretnie warstwy w CNN są odpowiedzialne za wykrywania wzorców w obrazach. Dla każdej z tych warstw należy określić ilość filtrów, które są potrzebne w danej warstwie. W zależności jaki filtr jest użyty, można wykryć różne wzorce, m.in. krawędzie, okręgi, tekstury czy całe obiekty. Filtry, które wykrywają proste kształty są zlokalizowane w początkowych warstwach. Im głębiej, tym bardziej zaawansowane filtry, do bardziej skomplikowanych kształtów, są używane. dlatego głębsze są w stanie odnaleźć w obrazie konkretne obiekty takie jak oczy, nos, uszy, włosy, łuski, pióra czy dziób. Filtry to nic innego jak macierze z odpowiednimi wartościami, które zostają przepuszczone przez obraz i generują tzw. mapy cech. Aby obliczyć wartość cechy w macierzy(i,j) w k-tej mapie cech l-tej warstwy, $z^l_{i,j,k}$

$$z^l_{i,j,k} = \mathrm{w}_k^{l\,T} \mathrm{x}^l_{i,j} + b^l_k \tag{2.1}$$

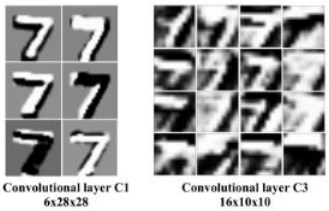
[10], gdzie w^l_k wektor wagi b^l_k bias k-tego filtra l-tej warstwy $\mathrm{x}^l_{i,j}$ łatka danych wejściowych wyśrodkowanych w (i,j) l-tej warstwy. Filtr w^l_k jest współdzielony, dzięki czemu model jest mniej złożony i sieć łatwiej się uczy. [10] Funkcje aktywacji, dzięki którym uzyskuje się nieliniowość sieci neuronowych, wprowadzają możliwość detekcji nieliniowych cech.

$$a^l_{i,j,k} = a(z^l_{i,j,k}) \tag{2.2}$$

Typowymi funkcjami aktywacji są sigmoida, tanh, ReLU.

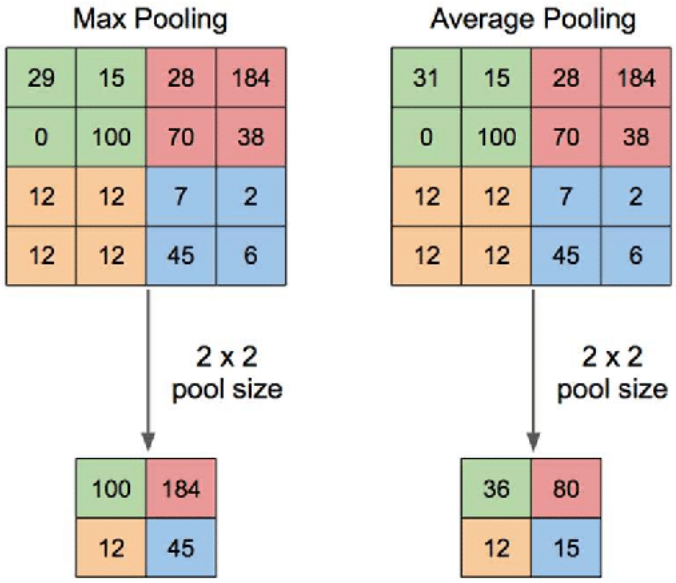
2.2.2 Warstwa redukująca (pooling)

Zadaniem warstwy redukującej jest, jak sama nazwa wskazuje, redukcja danych (zmniejszenie rozdzielczości mapy cech), aby usprawnić i przyspieszyć działanie programu. Zwykle warstwa ta jest położona między dwoma warstwami konwolucyjnymi. Każda mapa cech warstwy redukującej jest połączona z odpowiadającą jej mapą cech z poprzedniej warstwy konwolucyjnej.



Rys. 2.1: Przygotowanie zdjęcia

Najczęściej spotykaną operacją poolingu jest average pooling i max pooling. Na rys. * przedstawiono jak działają powyższe operacje. Rysunek * przedstawia mapę cech dla obrazu liczby 7 powstałej po uczeniu dwóch sieci konwolucyjnych. Filtry w pierwszej warstwie zostały zaprojektowane do wykrywania krawędzi, a w drugiej do wykrywania bardziej abstrakcyjnych cech. Dzięki dodawaniu kolejnych warstw konwolucyjnych i redukujących, można stopniowo zwiększać abstrakcyjność wyekstrahowanych cech. [10]



Rys. 2.2: Operacje poolingu

2.2.3 Warstwa spłaszczająca

Przedostatnią warstwą, przed tradycyjna siecią neuronową, której zadaniem jest klasyfikacja danego obiektu, jest warstwa spłaszczając. Jej zadaniem jest "spłaszczenie" wszystkich map cech do pojedynczego wektora, aby można było podać te dane na wejście kolejnej warstwy sztucznej sieci neuronowej.

2.2.4 Warstwa wyjścia

Zadaniem ostatniej warstwy jest klasyfikacja. Wektory wartości map cech zostaną podane na wejście tejże warstwy i na podstawie tych cech sieć uczy się i znajduje powiązania między nimi a klasą, do której ma być przyporządkowana.

3. Algorytm

4. Zastosowanie konwolucyjnych sieci neuronowych do rozpoznawania twarzy

4.1 Przykład 1

Jako pierwsza zostanie przedstawiona architektura AlexNet zaproponowana w [11]. Nie jest ona zaprojektowana ściśle pod rozpoznawanie twarzy, jednak jest to jedna z bardziej znanych architektur do rozpoznawania obiektów na obrazie.

4.1.1 Baza danych

Bazą danych na której oparte było uczenie sieci AlexNet była ImageNet. Składa się ona z 15 milionów wysokiej jakości zdjęć podzielonych na około 22 tysięcy klas. Została stworzona w oparciu o ekstrakcję danych z Internetu oraz skatalogowana przez użytkowników. W bazie znajdują się zdjęcia o różnych rozdzielczościach. AlexNet wymaga obrazów o rozmiarach 256x256, dlatego też zostało zastosowane przeskalowanie wszystkich zdjęć do takiej rozdzielczości. Wstępnie dane nie były przetwarzane w jakikolwiek sposób, z wyjątkiem odjęcia średniej aktywności z każdego piksela w zestawie treningowym. Sieć została nauczona w zakresie wyśrodkowanych, surowych wartości RGB pikseli.

4.1.2 Uczenie sieci

Uczenie sieci wymagało dużej mocy obliczeniowej, dlatego zostały wykorzystane do tego celu dwa procesory graficzne (GTX 580). Było to możliwe dzięki zastosowaniu równoległego połączenia dwóch procesorów, w których każdy z nich posiadał połowę neuronów. Równoległe połączenie sprawiało, że dane mogły być przesyłane bezpośrednio z jednego do drugiego procesora. Dodatkowo zostało jeszcze wprowadzone ograniczenie komunikacji między procesorami; mogły przesyłać dane tylko w określonych warstwach. Oznaczało to na przykład, że filtry z warstwy 3 otrzymywały dane wejściowe ze wszystkich map cech uzyskanych w warstwie 2. Jednakże neurony z warstwy 4 otrzymywały dane z warstwy 3 tylko z tych neuronów, które znajdowały się na tych samych procesorach. Wybranie odpowiedniej architektury połączeń dla dwóch procesorów jest dosyć kłopotliwe jeśli, ale takie rozwiązanie daje możliwość precyzyjnego dopasowania ilości komunikacji do dostępnej mocy obliczeniowej komputera.

Sieć została nauczona z użyciem stochastycznego spadku gradientu z grupą 128 przykładów, pędem = 0.9 i rozpadem wagi = 0.0005. Mała wartość rozpadu wagi jest bardzo ważna dla nauki modelu - redukuje błąd uczenia. Wagi aktualizuje się zgodnie ze wzorem:

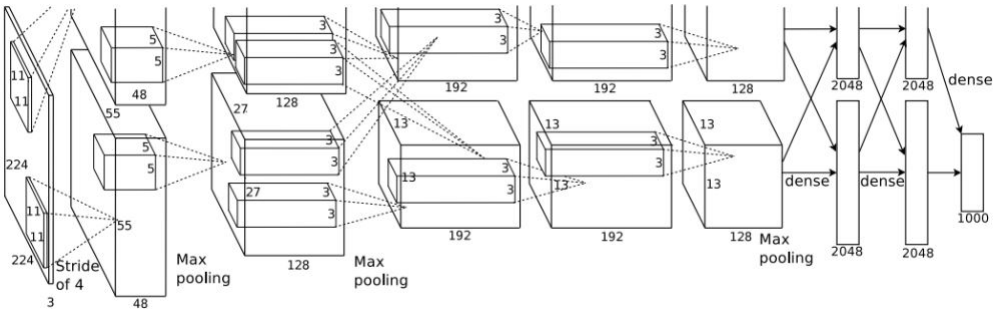
$$\begin{aligned} v_{i+1} &:= 0.9 \cdot v_i - 0.0005 \cdot \epsilon \cdot \omega_i - \epsilon \cdot \left\langle \frac{\partial L}{\partial \omega}|_{\omega_i} \right\rangle_{D_i} \\ v_{i+1} &:= 0.9 \cdot v_i - 0.0005 \end{aligned} \tag{4.1}$$

gdzie i -indeks iteracji, v -zmienna pędu, ϵ jest wskaźnikiem nauki, a $\left\langle \frac{\partial L}{\partial \omega}|_{\omega_i} \right\rangle_{D_i}$ jest średnią i-tej grupy D_i pochodnej celu. Wagi każdej warstwy zostały zainicjalizowane z zerowego średniego rozkładu Gaussa z odchyleniem standardowym = 0.01. Zainicjalizowane zostały również biasy neuronów zarówno w drugiej, czwartej i piątej konwolucyjnej warstwie, jaki i wszystkich warstwach spłaszczających ze stałą równą 1. Nadanie takich wartości początkowych przyspiesza wczesne etapy nauki zapewniając ReLu pozytywne dane wejściowe. Wartości początkowe biasów pozostałych warstw zostały ustawione na 0. Sieć została nauczona po około 90 cyklach z użyciem 1.2 miliona obrazów, co zajęło od pięciu do sześciu dni na dwóch procesorach NVIDIA GTX 580 3GB.

4.1.3 Architektura

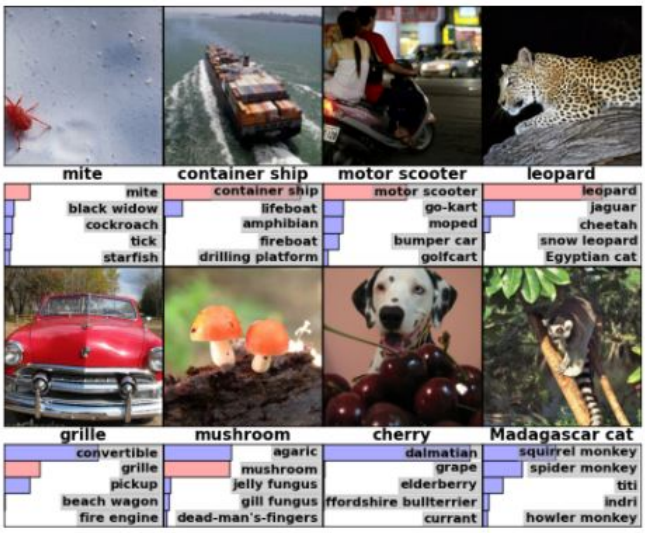
Sieć składa się z ośmiu warstw - pięć pierwszych warstw to warstwy konwolucyjne, a pozostałe trzy to warstwy spłaszczające. Wyjście ostatniej w pełni połączonej warstwy jest podawane do 1000-kierunkowej transformaty Softmax, która przyporządkowuje dane do ponad 1000 klas. Neurony drugiej, czwartej i piątej konwolucyjnej warstwy są połączone tylko do tych neuronów poprzednich warstw, które znajdują się na tym samym procesorze. Neurony trzeciej warstwy konwolucyjnej połączone są ze wszystkimi neuronami drugiej warstwy. Neurony warstw spłaszczających są połączone ze wszystkimi neuronami warstw poprzednich. Warstwy normalizacji odpowiedzi znajdują się zaraz po pierwszej i drugiej warstwie konwolucyjnej. Warstwa max-poolingu umieszczona została za warstwą normalizacji odpowiedzi i piątą warstwą konwolucyjną. Do wyjścia każdej warstwy konwolucyjnej oraz spłaszczającej zastosowana jest funkcja aktywacji ReLU.

Pierwsza warstwa konwolucyjna przetwarza zadany obraz 224x224x3 uzywając 96 filtrów o wymiarach $11 \times 11 \times 3$ z krokiem 4 pikseli. Druga warstwa na wejście otrzymuje wyjście poprzedniej warstwy po uprzednim znormalizowaniu i redukcji (poolingu), a następnie filtruje obraz z użyciem 256 filtrów 5x5x48. Trzecia, czwarta i piąta warstwa są ze sobą bezpośrednio połączone. Trzecia warstwa posiada 384 filtry o wymiarach 3x3x256, czwarta posiada 384 filtry o wymiarach 3x3x192, a piąta - 265 filtrów 3x3x192. Każda z warstw spłaszczających posiada 4096 neuronów. Architektura sieci AlexNet została przedstawiona na rys.4.1.



Rys. 4.1: Architektura AlexNet

4.1.4 Klasyfikacja



Rys. 4.2: Wyniki klasyfikacji

Rys. 4.3: Wyniki klasyfikacji

4.1.5 Wyniki

4.2 Przykład 2

Drugim przykładem jest sieć zaproponowana w [12]. sieć ta jest zaprojektowana ściśle dla rozpoznawania twarzy.

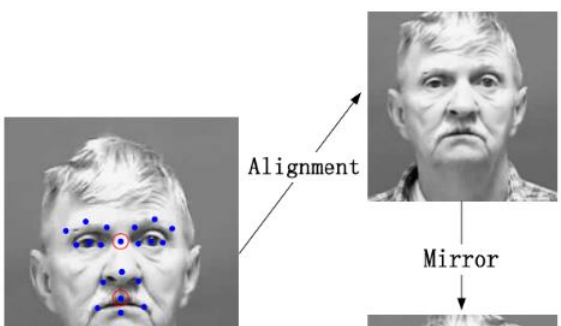
4.2.1 Baza danych

Bazą danych wykorzystaną do uczenia sieci została baza CASIA-WebFace, która została stworzona w Chinese Academy of Sciences (CASIA). Dane takie jak zdjęcie profilowe, imię i nazwisko oraz zdjęcia z galerii zostały pobrane z bazy danych aktorów ze strony IMDb. Pobrane dane nie nadawały się jeszcze jako baza do nauki sieci, dlatego należało je jeszcze przetworzyć i oznaczyć każdą z twarzy. Problem stanowiły zdjęcia z galerii, na których występowało więcej osób. Do przetworzenia zdjęć użyto metody grupowania podobieństwa znaczników, czyli porównywania zdjęć profilowych do wykrytych twarzy ze zdjęć z galerii wspomagając się tagami z nazwiskami dla każdego zdjęcia. Po przetworzeniu, baza została ręcznie zweryfikowana i poprawiona. Ostatecznie w sieci znajduje się 10,575 nazwisk i 494,414 zdjęć twarzy.

4.2.2 Uczenie sieci

Przed procesem uczenia, każdy obraz zostaje przekonwertowany do skali szarości i znormalizowany do rozmiaru 100×100 według dwóch punktów charakterystycznych zaznaczonych na obrazku.

Po normalizacji dystans pomiędzy tymi dwoma punktami wynosi 25 pikseli. Ponieważ twarz jest prawie symetryczna, można



podwoić zestaw treningowy poprzez odbicie lustrzane każdego zdjęcia. Dzięki temu zapewniona jest różnorodność pozy. Z taką ilością danych zmniejsza się prawdopodobieństwo do przeuczenia sieci, dlatego też rozpad wagi został ustawiony na 0 dla wszystkich warstw konwolucyjnych i na 0.0005 dla warstw w pełni połączonych. Współczynnik uczenia jest początkowo ustawiony na 0.01, po czym stopniowo maleje do 0.00001. Ponieważ współczynnik zbieżności sieci Softmax jest szybszy niż funkcja kosztów sieci Contrastive, waga jest początkowo ustawiona na małą wartość 0.00032, by później stopniowo ją zwiększać do wartości 0.0064. Do uczenia sieci została zastosowana publiczna sieć cuda-convnet [13]. Dla kosztów sieci Softmax wystarczą jedynie zdjęcia twarzy i ich podpisy, natomiast dla kosztów sieci Contrastive należy wygenerować pary twarzy przez pobrane próbki z zestawu treningowego. Aby ograniczyć zużycie pamięci i miejsca na dysku spróbkowano pozytywne i negatywne pary twarzy w każdej partii online.[14]

4.2.3 Architektura

Architektura głębokiej sieci konwolucyjnej składa się z połączenia kilku rozwiązań z ostatnich udanych sieci zawierających bardzo głęboką architekturę [14], reprezentacje nisko wymiarową i liczne funkcje strat (loss function) [15]. Małe rozmiary filtrów oraz głęboka architektura zmniejsza liczbę parametrów i zwiększa nieliniowość sieci. Niskopoziomowa reprezentacja jest zgodna z założeniem, że obrazy twarzy zwykle leżą na wielowymiarowej rozmaitości matematycznej i niskowymiarowe ograniczenia mogą zmniejszyć złożoność sieci. Łącząc identyfikację i weryfikację funkcji strat zostało przeanalizowane w [15], które mogą nauczyć się więcej reprezentacji dyskryminujących niż tylko sam Softmax.

Rozmiar wejściowej warstwy to 100x100x1 kanał, np. szary obraz. Zaproponowana sieć zawiera 10 warstw konwolucyjnych, 5 warstw poolingu i 1 w pełni połączoną warstwę. Więcej szczegółów dotyczących architektury pokazuje tabela.

Nazwa	Typ	Rozmiar filtra /krok	Rozmiar danych wyjściowych	Głębokość	#Parametry
Conv11	konwolucyjny	3×3 / 1	$100 \times 100 \times 32$	1	0.28K
Conv11	konwolucyjny	3×3 / 1	$100 \times 100 \times 64$	1	18K
Pool1	maksimum	2×2 / 2	$50 \times 50 \times 64$	0	
Conv21	konwolucyjny	3×3 / 1	$50 \times 50 \times 64$	1	0.28K
Conv22	konwolucyjny	3×3 / 1	$50 \times 50 \times 128$	1	18K
Pool2	maksimum	2×2 / 2	$25 \times 25 \times 128$	0	
Conv31	konwolucyjny	3×3 / 1	$25 \times 25 \times 96$	1	0.28K
Conv32	konwolucyjny	3×3 / 1	$25 \times 25 \times 192$	1	18K
Pool3	maksimum	2×2 / 2	$13 \times 13 \times 192$	0	
Conv41	konwolucyjny	3×3 / 1	$13 \times 13 \times 128$	1	0.28K
Conv42	konwolucyjny	3×3 / 1	$13 \times 13 \times 256$	1	18K
Pool4	maksimum	2×2 / 2	$7 \times 7 \times 256$	0	
Conv51	konwolucyjny	3×3 / 1	$7 \times 7 \times 160$	1	0.28K
Conv52	konwolucyjny	3×3 / 1	$7 \times 7 \times 320$	1	18K
Pool5	średnia	7×7 / 1	$1 \times 1 \times 320$	0	
Dropout	redukcja połączeń (40%)		$1 \times 1 \times 320$	0	
Fc6	w pełni połączone		10575	1	3305K
Cost1	softmax		10575	1	
Cost2	kontrastujący		1	0	
Suma				11	5015K

Rozmiar filtrów to 3×3. Pierwsze cztery wykorzystują max-pooling (funkcje maksimum z obszaru filtru), ostatnia wykorzystuje average-pooling (średnią z obszaru filtru). Architektura nie jest w pełni optymalna przez ograniczenie mocy obliczeniowej procesora. Małe filtry i bardzo głęboka architektura została zaproponowana w [14] i [16]. [16] osiągnął wysokie wyniki w konkursie ImageNet 2014 dzięki 19-warstwowej sieci. W międzyczasie, [16] osiągnął trochę lepsze rezultaty niż [14] dzięki 22-warstwowej sieci. Ta architektura zawiera w sobie rozwiązania z obu tych sieci. Użyte zostały wiele małych filtrów do aproksymacji większych filtrów i usunięcia nadmiarowych w pełni połączonych warstw aby zmniejszyć liczbę parametrów. Ostatecznie sieć używa filtrów 3×3 we wszystkich warstwach konwolucyjnych i posiada tylko jedną warstwę w pełni połączoną.

4.2.4 Klasyfikacja

Klasyfikacja została przeprowadzona na dwóch zestawach danych: LFW (Life Faces in the Wild) i YTF (YouTube Faces). Istnieją trzy protokoły do raportowania wydajności LFW: protokół bez nadzoru, ograniczony i nieograniczony. Do oceny wyników przedstawienia twarzy używa się protokołu bez nadzoru, a pozostałych dwóch używa się do oceny uczenia metrycznego lub do oceny całej metody. Dla wszystkich protokołów zestaw uczący jest ściśle określony - zawiera 6000 par twarzy w 10 grupach. Wyniki przedstawia tabela:

Zbiór testowy YTF został użyty aby przetestować zdolność sieci do generalizacji. Wyniki

przedstawia tabela2.

4.2.5 Wyniki

Metoda	Sieć	Dokładność ± BS	Protokół
DeepFace	1	95.92 ± 0.29 %	bez nadzoru
DeepFace	1	97.00 ± 0.28 %	ograniczony
DeepFace	3	97.15 ± 0.27 %	ograniczony
DeepFace	7	97.35 ± 0.25 %	nieograniczony
DeepID2	1	95.43 %	nieograniczony
DeepID2	2	97.28 %	nieograniczony
DeepID2	4	97.75 %	nieograniczony
DeepID2	25	98.97 %	nieograniczony
A	1	96.13 ± 0.30 %	bez nadzoru
B	1	96.30 ± 0.35 %	bez nadzoru
C	1	97.30 ± 0.31 %	bez nadzoru
D	1	96.33 ± 0.42 %	bez nadzoru
E	1	97.73 ± 0.31 %	nieograniczony

gdzie

- A: DR + Cosine;
- B: DR + PCA on CASIA-WebFace + Cosine;
- C: DR + Joint Bayes on CASIA-WebFace;
- D: DR + PCA on LFW training set + Cosine;
- E: DR + Joint Bayse on LFW training set.

Metoda	Sieć	Dokładność ± BS	Protokół
DeepFace	1	91.4 ± 1.1 %	z nadzorem
A	1	88.00 ± 1.5 %	bez nadzoru
B	1	90.60 ± 1.24 %	bez nadzoru
C	1	92.24 ± 1.28 %	z nadzorem

- A: DR + Cosine;
- D: DR + PCA on YTF training set + Cosine;
- E: DR + Joint Bayes on YTF training set.

4.3 Przykład 3

W [17] została przedstawiona sieć do rozpoznawania twarzy w czasie rzeczywistym (np. w monitoringu). Zostało to osiągnięte poprzez zastosowanie bazy danych z różnymi zdjęciami tej samej osoby (z różną pozą, wyrazem twarzy).

4.3.1 Baza danych

Baza danych, która została użyta dla tej sieci to baza ORL. Zawiera ona 10 zestawów zdjęć od 40 różnych osób. Niektóre z nich zostały zrobione w różnym czasie. Różnorodność bazy zapewniają wariacje w obrębie tej samej osoby - otwarte/zamknięte oczy, uśmiech/bez uśmiechu, z okularami i bez. Wszystkie zdjęcia zostały zrobione na ciemnym, jednorodnym tle w pozycji en face z tolerancją odchylenia lub rotacji o 20 stopni. Wszystkie obrazy są w skali szarości o rozdzielczości 92×112.

4.3.2 Uczenie sieci

Przed uczeniem każdy obraz został poddany próbkowaniu. Polega ono na tym, że każdą próbkę otrzymuje się nakładając na oryginalny obraz okna o wymiarach np. 5×5, a następnie wyodrębnia się z tego obszaru jeden piksel, który znajdzie się w obrazie wynikowym. Potem okno jest przesuwane o 4 piksele i powtarza się poprzednie kroki aż do przepróbkowania całego obrazu. Następnie następuje proces uczenia za pomocą map samoorganizujących (np. z trzema wymiarami i pięcioma węzłami na wymiar co daje w sumie $5^3 = 125$ węzłów). Mapa jest trenowana na wektorach otrzymanych z poprzedniego etapu. Samoorganizująca mapa kwantyzuje 25-wymiarowy wektor wejściowy do 125 wartości uporządkowanych topologicznie. Trzy wymiary mapy można traktować jako trzy cechy. Został również przeprowadzony eksperyment, gdzie zastąpiono samoorganizujące mapy na transformatę Karhunena-Loévego. W takim przypadku, transformata rzutuje wektory z 25-wymiarowej przestrzeni na przestrzeń trójwymiarową. Kolejnym krokiem jest ponowne użycie okna na zestawie uczącym i testowym. Lokalne próbki obrazu są przekazywane do mapy po każdym kroku, tworząc w ten sposób nowe zestawy uczące i testowe w przestrzeni wyjściowej stworzonej przez samoorganizujące się mapy. Powstałe obrazy wejściowe są reprezentowane przez 3 mapy, gdzie każda z nich odpowiada wymiarom w samoorganizujących mapach. Rozmiar tych map jest równy rozmiarom obrazów (92×112) podzielonych przez rozmiar kroku okna (w tym wypadku dla kroku 4, mapy maja rozmiar 23×28) Sieć konwolucyjna jest uczona na nowo powstałym zestawie testowym.

4.3.3 Architektura

Sieć konwolucyjna składa się z pięciu warstw, nie wliczając warstwy wyjściowej. Dla każdej klasyfikacji została policzona ocena wiarygodności: $y_m(y_m - y_{2m})$, gdzie y_m jest maksimum wyjścia, y_{2m} jest drugim maksimum wyjścia (dla wyjść które zostały poddane transformacie softmax:

$$y_i = \frac{\exp(u_i)}{\sum_{j=1}^{k} \exp(u_j)}, \tag{4.2}$$

gdzie u_i-oryginalny wektor wyjściowy, y_i wynik transformaty, k- ilość danych wyjściowych). Sieć została poddana uczeniu z wsteczną propagacją błędu w sumie dla 20 000 aktualizacji. Dokładną architekturę sieci zawarto w tabeli:

Warstwa	Typ	Jednostka	x	y	Pole recep. x	Pole recep. y	Połączenia (%)
1	Konwolucyjny	20	21	26	3	3	100
2	Podpróbkujący	20	9	11	2	2	-
3	Konwolucyjny	25	9	11	3	3	30
4	Podpróbkujący	25	5	6	2	2	-
5	W pełni połączony	40	1	1	5	6	100

Tab. 4.1: Architektura sieci

Połączenia(%) informują o połączeniach węzłów z poprzednią warstwą, wartość mniejsza niż 100% redukuje ilość wag w sieci i może ulepszyć zdolność do generalizacji sieci.

4.3.4 Klasyfikacja

Zostało przeprowadzonych kilka eksperymentów. Dla każdego z nich było przygotowanych pięć obrazów do uczenia i pięć obrazów testowych dla każdej osoby, co dało w sumie 200 obrazów do uczenia i 200 obrazów testowych. Żadne zdjęcia nie powtarzały się w obydwóch zestawach. Stałe wartości dla każdego eksperymentu:

- liczba klas: 40,
- metoda redukcji wymiarów: mapy samoorganizujące,
- wymiar map samoorganizujących: 3,
- ilość węzłów na każdy wymiar mapy: 5,
- ilość obrazów do uczenia na klasę: 5.

4.3.5 Wyniki

Tabele dla każdego eksperymentu pokazują współczynnik błędu przy zmiennych, badanych wartościach. Wyniki są średnią z 3 symulacji.

1. Wpływ wymiaru mapy samoorganizującej - badanie dla wymiarów od 1 do 4. Najlepszy wynik dla mapy trójwymiarowej.

Wymiar	1	2	3	4
Współczynnik błędu	8.25%	6.75%	5.75%	5.83%

2. Wpływ poziomu kwantyzacji mapy samoorganizującej - badanie dla mapy trójwymiarowej od 4 do 8 węzłów na każdy wymiar. Najlepszy wynik uzyskano dla 8 węzłów. Jest to też najlepszy wynik wśród wszystkich przeprowadzonych eksperymentów.

Poziom	4	5	6	7	8
Współczynnik błędu	8.5%	5.75%	6.0%	5.75%	3.83%

3. Zastąpienie map samoorganizujących na transformatę Karhunena-Loévego. Najlepszy wynik dały mapy samoorganizujące.

Redukcja wymiarów	Dyskretna transformata KL	SOM
Współczynnik błędu	5.33%	3.83%

4. Zastąpienie konwolucyjnej sieci neuronowej wielowarstwowym perceptronem. Najlepszy wynik dały mapy samoorganizujące.

	Dyskretna transformata KL	SOM
WWP	41.2%	39.6%
CN	5.33%	3.83%

5. Wpływ ilości zdjęć jednej osoby używanych w zestawie treningowym (a - średnia na klasę, b - jeden na obraz). Najlepszy wynik dla sieci konwolucyjnej z SOM z użyciem pięciu zdjęć.

Zdjęcia na osobę	1	2	3	4	5
Eigenfaces(a)	38.6%	28.8%	28.9%	27.1 %	26%
Eigenfaces(b)	38.6%	20.9%	18.2%	15.4 %	10.5 %
DT-KL+CN	34.2 %	17.2 %	13.2 %	12.1 %	7.5%
CN+SOM	30.0 %	17.0%	11.8 %	7.1%	3.5%

5. Wady, zalety, udoskonalenia

Bibliografia

[1] Bledsoe, W.W,*The model method in facial recognition* Panoramic Research Inc., Palo Alto, CA, Rep. PRI:15, August 1966

[2] Bledsoe, W.W,*Man machine facial recognition* , Panoramic Research Inc., Palo Alto, CA, Rep. PRI:22, August 1996

[3] L. Sirovich and M. Kirby, *Low-Dimensional procedure for the characterization of human faces.* Journal of optical society of America Vol 4 page 519 March 1987

[4] Mattew A Turk and Alex P. Pentland, *Recognition using Eigen faces, vision and modeling group*, The media laboratory , Massachusetts Institute of Technology, 1991

[5] Jonathon Phillips, Patrick J. Rauss, and Sandor Z. De, FERET (Face Recognition Technology) Recognition Algorithm Development and Test Results, Army Research Laboratory (ARL), October 1996

[6] P. Jonathon Phillips, Patrick J. Flynn Todd Scruggs Kevin W. Bowyer, William Worek, Preliminary Overview of the Face Recognition Grand Challenge, IEEE Conference on Computer Vision and Pattern Recognition 2005

[7] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, Deep Face Recognition, Visual Geometry Group, Department of Engineering Science, University of Oxford

[8] West, J (2017) History of Face Recognition – Facial recognition software [online] FaceFirst Face Reconition facial recognition software available on https://www.facefirst.com/blog/brief-of-face recognition-software/ [Accessed 15 Oct. 2018

[9] Carey, S., and Diamond, R, "From Piecemeal to Configurational Representation of Faces", Science 195, pp.312 313, (1977).

[10] Bledsoe, W.W,*"Man machine facial recognition"* , Panoramic Research Inc., Palo Alto, CA, Rep. PRI:22, August 1996

[11] Bledsoe, W.W,*"Man machine facial recognition"* , Panoramic Research Inc., Palo Alto, CA, Rep. PRI:22, August 1996

[12] Bledsoe, W.W,*"Man machine facial recognition"* , Panoramic Research Inc., Palo Alto, CA, Rep. PRI:22, August 1996

[13] Bledsoe, W.W,*"Man machine facial recognition"* , Panoramic Research Inc., Palo Alto, CA, Rep. PRI:22, August 1996