

Rozdział 5

Porównanie skuteczności algorytmów rozpoznawania twarzy w zależności od poziomu zniekształcenia zdjęcia

Daniel Grabowski*

Streszczenie. Niniejsze badanie analizuje wpływ zniekształceń zdjęć na skuteczność algorytmów rozpoznawania twarzy, koncentrując się na konwolucyjnych sieciach neuronowych (CNN) ze względu na ich wysoką efektywność. Poprzez przegląd istniejących rozwiązań i eksperymenty z wykorzystaniem bazy zdjęć LFW, przeanalizowano wpływ kompresji, rozmycia i szumu na dokładność działania algorytmów. Eksperymenty przeprowadzono z użyciem biblioteki DeepFace w języku Python, porównując podobieństwo zdjęć i optymalizując parametry za pomocą analizy gęstości i drzew decyzyjnych. Badanie wykazało, że kompresja w niewielkim stopniu wpływa na skuteczność, natomiast szum – szczególnie przy wyższym poziomie – znacząco obniża dokładność. Zastosowanie filtrów medianowych do zdjęć z szumem w dużej mierze przywraca ich efektywność. Wpływ rozmycia zależy od kierunku – rozmycie pionowe istotnie pogarsza skuteczność rozpoznawania. Spośród analizowanych algorytmów, FaceNet512 wykazał się największą odpornością na różne typy zniekształceń. Wyniki podkreślają znaczenie rodzaju zniekształcenia i optymalizacji parametrów dla utrzymania wysokiej dokładności, oferując cenne wskazówki dla zwiększenia odporności algorytmów na pogorszoną jakość zdjęć.

Słowa kluczowe: rozpoznawanie twarzy, uczenie głębokie, konwolucyjne sieci neuronowe, kompresja, zniekształcenia obrazu

5.1. Wprowadzenie

Ludzkie twarze odgrywają kluczową rolę w komunikacji i identyfikacji, niosąc za sobą znaczenie ewolucyjne poprzez przekazywanie istotnych informacji o innych osobach, takich jak płeć, rasa, stan emocjonalny czy atrakcyjność. Wyraz twarzy stanowi również istotne źródło komunikacji niewerbalnej, a umiejętność interpretacji twarzy i rozpoznawania członków rodziny, przyjaciół czy wrogów pojawia się już

* Wydział Informatyki, Politechnika Białostocka, Białystok, Poland, daniel.grabowski@pb.edu.pl
DOI 10.24427/978-83-68673-08-1_5

we wczesnym dzieciństwie. Ta wrodzona zdolność jest podstawą codziennych interakcji społecznych, umożliwiając rozpoznawanie emocji otoczenia oraz identyfikację znajomych w tłumie.

Historia biometrycznych systemów rozpoznawania twarzy sięga lat 60. XX wieku – początkowo opierały się one na geometrii twarzy, lecz z biegiem lat ewoluowały dzięki wzrostowi mocy obliczeniowej, rozwojowi algorytmów oraz dostępności wysokiej jakości baz obrazów. Współcześnie systemy te są powszechnie stosowane w celach bezpieczeństwa, m.in. przy weryfikacji tożsamości na lotniskach czy identyfikacji osób zaginionych i ciał.

Pomimo szerokiego zastosowania i rosnącej obecności w życiu codziennym, algorytmy rozpoznawania twarzy – szczególnie te oparte na głębokich sieciach neuronowych – napotykają trudności związane z jakością danych. Zniekształcenia zdjęć wpływają negatywnie na precyzję algorytmów, co jest hipotezą badawczą analizowaną w niniejszej pracy.

5.2. State of the Art

Pierwsze biometryczne systemy rozpoznawania twarzy zaczęły pojawiać się w latach 60. XX wieku. Najwcześniejsze algorytmy opierały się głównie na geometrii twarzy, porównując odległości między charakterystycznymi punktami, takimi jak kąciaki ust czy środek oczu. Metody te wymagały jednak ręcznego wyznaczania położenia tych punktów, co czyniło je niezwykle pracochłonnymi. Dzisiejsze systemy wyróżniają się pełną automatyzacją wszystkich etapów działania systemu rozpoznawania twarzy, tj. wykrywania twarzy, ekstrakcji cech i samego rozpoznawania. Najbardziej zaawansowane algorytmy osiągają bardzo wysoką dokładność klasyfikacji, sięgającą nawet powyżej 99%; jednakże istnieją czynniki, które negatywnie wpływają na ten poziom skuteczności. Można je podzielić na czynniki wewnętrzne, związane ze zmianami w wyglądzie osoby, oraz zewnętrzne, związane z jakością obrazu ?.

5.2.1. FaceNet

FaceNet to system opracowany przez firmę Google, który uczy się odwzorowywać obrazy twarzy bezpośrednio w zwartą przestrzeń euklidesową, w której odległości odpowiadają miarze podobieństwa twarzy. W podstawowej wersji odwzorowanie zachodzi do przestrzeni 128-wymiarowej, jednak istnieje również wariant, w którym odwzorowanie odbywa się do przestrzeni 512-wymiarowej. W badaniu opartym na bazie LFW, FaceNet osiągnął skuteczność na poziomie 99,63%.

W eksperymentach system był trenowany z wykorzystaniem metody stochastycznego spadku gradientu (SGD) i standardowego algorytmu propagacji wstecznej przez 1000–2000 godzin na klastrze komputerów. Model wykorzystywał warstwy konwolucyjne o rozmiarach $1 \times 1 \times d$ pomiędzy standardowymi warstwami konwolucyjnymi w celu zwiększenia wydajności, co zaowocowało modelem o głębokości 22 warstw ?.

5.2.2. VGG-Face

Algorytm VGG-Face został stworzony na Uniwersytecie Oksfordzkim w 2015 roku. Autorzy postawili sobie dwa cele: pierwszy to stworzenie metody umożliwiającej tworzenie dużych zbiorów zdjęć do uczenia maszynowego. Drugim celem było zbadanie różnych architektur konwolucyjnych sieci neuronowych, w tym sposobów normalizacji zdjęć oraz metod uczenia metryk. Autorzy przeanalizowali stosowane modele rozpoznawania twarzy, aby wybrać najlepsze praktyki i wyeliminować mniej istotne szczegóły, co pozwoliło osiągnąć prostszą strukturę sieci neuronowej przy zachowaniu skuteczności najlepszych rozwiązań na rynku. W porównaniu do algorytmów takich jak DeepFace i DeepID, zastosowano dwuwymiarowe rozpoznawanie geometrii twarzy zamiast trójwymiarowego ?.

5.2.3. ArcFace

ArcFace to model głębokiej konwolucyjnej sieci neuronowej, który cechuje się małą odległością wewnątrz klas decyzyjnych i dużą odległością między klasami. Osiągnięto to dzięki opracowaniu algorytmu Additive Angular Margin Loss (ArcFace), który w tym modelu zastępuje często wykorzystywany algorytm softmax. Znacząco poprawia to zdolność dyskryminacyjną modelu i stabilizuje proces uczenia. W eksperymentach system ArcFace został wytrenowany na bazie ponad 5 milionów obiektów. Algorytm ArcFace osiągnął skuteczność 99,83% na bazie LFW ?.

Zidentyfikowano wady znanych metod i podjęto próbę ich uzupełnienia poprzez własne podejście. Autorzy zdecydowali się na usprawnienie metodologii oraz poprawę skuteczności działania algorytmów głębokiego uczenia w kontekście rozpoznawania twarzy ludzkich. Tym samym uzupełniono lukę badawczą dotyczącą wpływu zniekształceń obrazu. Proponowane podejście przedstawiono w sekcji 4.

5.3. Tło badawcze

5.3.1. Wprowadzenie do wyboru danych

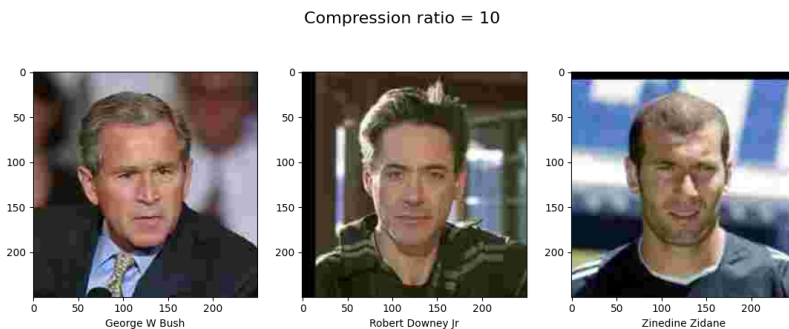
Wybór danych jest kluczowym krokiem w każdym projekcie badawczym, szczególnie w badaniach dotyczących technologii rozpoznawania twarzy. Polega on na wyborze zbiorów danych, które są reprezentatywne dla rzeczywistych scenariuszy, w których będą stosowane algorytmy. Zapewnia to, że wyniki badań będą wiarygodne i możliwe do zastosowania poza warunkami laboratoryjnymi.

5.3.2. Wybrany zbiór danych

Jedną z baz danych wykorzystanych w tym badaniu była baza LFW (Labeled Faces in the Wild). Została ona wybrana ze względu na fakt, że jest powszechnie stosowana jako punkt odniesienia w wielu innych badaniach oceniających skuteczność algorytmów rozpoznawania twarzy. Zawiera ponad 13 000 przyciętych zdjęć twarzy ludzkich. Baza obejmuje zdjęcia ponad 5700 osób, z czego 1680 osób ma więcej niż jedno zdjęcie.

5.3.3. Prezentacja danych

Na potrzeby badania utworzono zbiory danych zawierające zdjęcia zmodyfikowane przez różne zniekształcenia. Skuteczność algorytmów będzie oceniana w zależności od poziomu kompresji zdjęć, rozmycia oraz szumu. Kompresja została wykonana na trzech poziomach: 50, 20 i 10. Rozmycie podzielono na poziome i pionowe, przy czym każde wykonano z użyciem filtrów o rozmiarze okna równym 5 i 10. Ostatnie bazy danych zostały utworzone ze zdjęć, do których zastosowano szum Gaussa, szum typu sól i pieprz (S&P) oraz obrazy odszumione za pomocą filtra medianowego.



Rysunek 5.1: Skompresowane zdjęcia, współczynnik kompresji = 10

5.4. Proponowane podejście

Do przeprowadzenia eksperymentów wybrano bibliotekę DeepFace napisaną w języku Python. Udostępnia ona gotowe modele wielu dostępnych na rynku algorytmów rozpoznawania twarzy oraz możliwość pobrania i załadowania wytrenowanych wag sieci neuronowych, uczonych na milionach zdjęć. Biblioteka umożliwia również analizę atrybutów takich jak wiek, rasa czy nastrój osoby na zdjęciu. Aby modele działały jak najlepiej, konieczne jest ustalenie progów dla każdego z algorytmów.

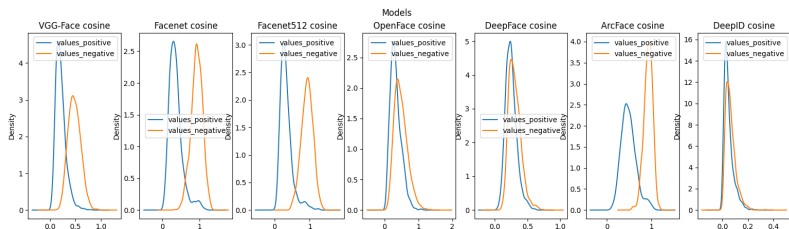
5.4.1. Metryka odległości wektorów

W eksperymentach porównywane będą wektory cech zwracane przez badane algorytmy. Każdy algorytm przyjmuje na wejściu zdjęcie osoby i zwraca wektor cech. Odległości między wektorami uzyskanymi z dwóch zdjęć będą porównywane i na tej podstawie klasyfikowane będzie, czy zdjęcia przedstawiają tę samą osobę, czy różne. Odległość między wektorami cech obliczana jest metodą cosinusową.

5.4.2. Ustalanie progów dla algorytmów

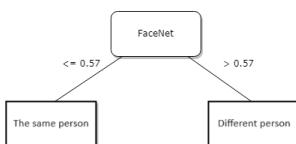
Aby zoptymalizować działanie algorytmów, konieczne jest ustalenie dla każdego algorytmu progu, od którego przypadki będą klasyfikowane jako pozytywne lub negatywne. W tym celu przeprowadzono analizę obliczonych odległości na niezmo-dyfikowanej bazie zdjęć, a następnie narysowano wykresy gęstości przypadków.

Na podstawie wykresu można zobaczyć, gdzie koncentrują się przypadki pozytywne i negatywne. Im większa odległość gęstości przypadków negatywnych od



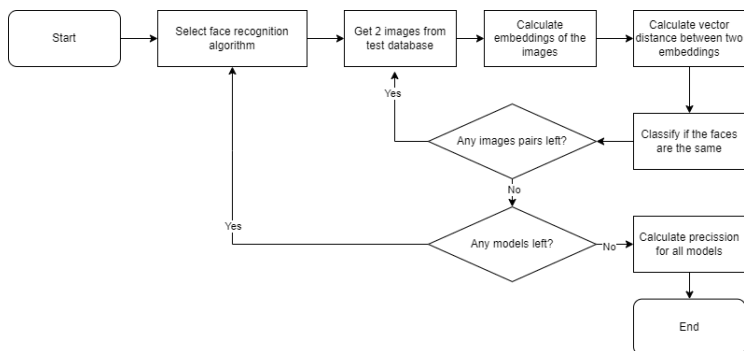
Rysunek 5.2: Rozkład przypadków pozytywnych i negatywnych dla wszystkich modeli

pozytywnych, tym większe różnice można oczekiwać przy porównywaniu zdjęć dwóch klas, co pozytywnie wpłynie na precyzję modelu. Ważne jest także, aby jak najwięcej przypadków koncentrowało się w jednym punkcie, tj. aby wykres gęstości był jak najwęższy i nie nachodził na wykres klasy przeciwnej.



Rysunek 5.3: Przykład drzewa decyzyjnego utworzonego dla algorytmu FaceNet

Następnie na podstawie tych danych utworzono drzewa decyzyjne, mające na celu znalezienie najlepiej dopasowanego progu. Na przykładzie drzewa decyzyjnego stworzonego dla algorytmu FaceNet: jeśli odległość cosinusowa między dwoma wektorami cech jest mniejsza lub równa 0,57, badane zdjęcia uznawane są za przedstawiające tę samą osobę. W przeciwnym razie – jeśli odległość jest większa – uznaje się, że osoby są różne.



Rysunek 5.4: Schemat blokowy działania algorytmu

5.4.3. Metoda wykrywania twarzy

Ważnym wyborem jest również algorytm wykrywania twarzy i przycinania zdjęcia. Biblioteka DeepFace implementuje algorytmy takie jak OpenCV, Dlib, MTCNN i RetinaFace. Algorytmy MTCNN i RetinaFace cechują się największą skutecznością, jednak działają najwolniej spośród wymienionych. Aby osiągnąć jak najlepsze wyniki kosztem czasu wykonania, wykorzystano algorytm MTCNN ???.

5.5. Eksperymenty

Aby określić, jak zniekształcenia zdjęć wpływają na wyniki, w pierwszej kolejności sprawdzono skuteczność działania algorytmów na bazie niezniszczonych zdjęć.

5.5.1. Eksperyment 1: Kompresja

Pierwszy eksperyment bada, jak poziom kompresji wpływa na skuteczność analizowanych algorytmów. Zdjęcia zostały skompresowane odpowiednio do poziomów 50%, 20% i 10%. Dodatkowo przeanalizowano, jak zmniejsza się rozmiar bazy zdjęć w porównaniu do oryginalnej oraz jakie oszczędności miejsca na dysku to przynosi. Zbadano również, ile czasu zajmuje przetwarzanie i porównywanie zdjęć w zależności od stopnia kompresji.

Tabela 5.1: Skuteczność modeli rozpoznawania twarzy przy różnych poziomach kompresji

Model	Normalne	Kompresja 50%	Kompresja 20%	Kompresja 10%
VGG-Face	88,6%	87,8%	87,2%	86,0%
FaceNet512	95,5%	94,7%	94,1%	92,3%
FaceNet	93,8%	93,3%	92,5%	91,2%
ArcFace	93,7%	93,7%	92,8%	89,6%

Kompresja na poziomie 50% nie wpłynęła znacząco na skuteczność testowanych algorytmów. Model ArcFace osiągnął ten sam wynik co na oryginalnej bazie, a pozostałe algorytmy odnotowały nieznaczny spadek dokładności klasyfikacji. Przy kompresji na poziomie 20% zauważono wyraźny spadek skuteczności dla wszystkich algorytmów – średnio o około 1,2 punktu procentowego. Ponownie model ArcFace wykazał najmniejszy spadek. Kompresja na poziomie 10% spowodowała większy spadek skuteczności we wszystkich algorytmach. ArcFace, który wcześniej wykazywał najmniejsze spadki, tym razem zanotował największy regres.

5.5.2. Eksperyment 2: Szum

Ten eksperyment symuluje zakłócenia, które mogą pojawić się na zdjęciach cyfrowych, np. w wyniku nagłej utraty sygnału lub jego osłabienia. Do zdjęć dodano szum Gaussa oraz szum typu sól i pieprz (S&P), który losowo zmieniał 5% i 10% pikseli. Następnie zastosowano filtr medianowy o rozmiarze okna równym 5 do zdjęć z szumem typu S&P, aby wygładzić zakłócenia i sprawdzić, w jakim stopniu algorytmy odzyskują skuteczność.

5.5.2.1. Szum Gaussa

Na początku sprawdzono wpływ dodania szumu Gaussa. Wszystkie algorytmy zanotowały spadek dokładności klasyfikacji. Reakcje algorytmów były różne: np. DeepFace wykazywał silną tendencję do klasyfikowania zdjęć jako różnych osób (wysoka czułość, ale niska precyzja), podczas gdy ArcFace odwrotnie – klasyfikował zdjęcia jako tę samą osobę, co skutkowało wysoką precyzją, ale niską ogólną trafnością.

Tabela 5.2: Precyzja modeli rozpoznawania twarzy z szumem Gaussa

Model	Normalne	Szum Gaussa
ArcFace	93,7%	74,6%
FaceNet	93,8%	85,2%
FaceNet512	95,5%	92,4%
VGG-Face	88,6%	83,3%

5.5.2.2. Szum S&P

W kolejnym etapie zbadano skuteczność algorytmów na bazie zdjęć, do których dodano szum typu sól i pieprz (S&P), polegający na losowym ustawieniu 5% lub 10% pikseli na wartości skrajne (biała czerń). Zdjęcia z tym typem szumu powodowały istotny spadek dokładności klasyfikacji. Wystąpiły również znane z wcześniejszych eksperymentów tendencje: np. ArcFace przy 10% szumu osiągnął bardzo wysoką precyzję (97,8%), ale efektywność spadła niemal do 50%, ponieważ algorytm zaczął klasyfikować większość par jako zdjęcia tej samej osoby. Podobne reakcje na szum zaobserwowano w algorytmach FaceNet i VGG-Face.

Po analizie skuteczności na zdjęciach z szumem, poddano je filtracji medianowej w celu usunięcia zakłóceń. Zastosowano filtr medianowy o rozmiarze okna równym 5, a następnie ponownie przeprowadzono eksperyment. Po filtracji algorytmy osiągnęły

skuteczność zbliżoną do tej z oryginalnej bazy, co potwierdziło skuteczność filtracji w odzyskiwaniu jakości klasyfikacji.

Tabela 5.3: Skuteczność modeli przy 5% szumie typu S&P i po filtrze medianowym

Model	Normalne	S&P 5%	Po filtrze medianowym
VGG-Face	88,6%	77,8%	86,2%
FaceNet512	95,5%	87,4%	92,2%
FaceNet	93,8%	80,0%	90,3%
ArcFace	93,7%	74,2%	91,9%

Tabela 5.4: Skuteczność modeli przy 10% szumie typu S&P i po filtrze medianowym

Model	Normalne	S&P 10%	Po filtrze medianowym
VGG-Face	88,6%	65,1%	85,3%
FaceNet512	95,5%	77,1%	93,3%
FaceNet	93,8%	60,8%	89,5%
ArcFace	93,7%	53,8%	89,9%

5.5.3. Eksperyment 3: Rozmycie

Celem eksperymentu było sprawdzenie, jak testowane algorytmy radzą sobie ze zdjęciami poddanymi rozmyciu. Zdjęcia rozmywano pionowo i poziomo, aby sprawdzić wpływ kierunku rozmycia. Oba rozmycia przeprowadzono przy użyciu filtrów o rozmiarze okna równym 5 i 10 pikseli.

5.5.3.1. Rozmycie poziome

W przypadku rozmycia poziomego zastosowanie filtra o rozmiarze 5 pikseli miało niewielki wpływ na jakość klasyfikacji. Użycie filtra o rozmiarze 10 pikseli miało nieco większy wpływ na skuteczność, lecz algorytmy nadal osiągały wysoką dokładność.

5.5.3.2. Rozmycie pionowe

Rozmycie pionowe filtrem o rozmiarze 5 pikseli wpłynęło podobnie jak poziome. Natomiast rozmycie pionowe filtrem o rozmiarze 10 pikseli spowodowało znaczny

spadek skuteczności dla wszystkich algorytmów – większy niż w przypadku rozmycia poziomego.

Tabela 5.5: Wpływ rozmycia poziomego na precyzję algorytmów rozpoznawania twarzy

Model	Normalne	Rozmycie 5	Rozmycie 10
VGG-Face	88,6%	87,1%	85,4%
FaceNet512	95,5%	93,3%	90,3%
FaceNet	93,8%	92,2%	86,8%
ArcFace	93,7%	92,9%	88,4%

Tabela 5.6: Wpływ rozmycia pionowego na precyzję algorytmów rozpoznawania twarzy

Model	Normalne	Rozmycie pionowe 5	Rozmycie pionowe 10
VGG-Face	88,6%	87,2%	77,5%
FaceNet512	95,5%	94,1%	82,6%
FaceNet	93,8%	88,4%	54,2%
ArcFace	93,7%	92,8%	66,7%

5.6. Analiza wyników

W przeprowadzonych eksperymentach zbadano wpływ zniekształceń zdjęć na skuteczność algorytmów rozpoznawania twarzy opartych na konwolucyjnych sieciach neuronowych. Analizie poddano wpływ kompresji, szumu oraz rozmycia zdjęcia.

Na początku sprawdzono skuteczność badanych algorytmów na bazie zdjęć LFW. Cztery algorytmy uzyskały wysokie wyniki: VGG-Face, FaceNet, FaceNet512 i ArcFace. Algorytmy OpenFace, DeepFace i DeepID uzyskały znacznie niższe wyniki, co mogło wynikać z niewłaściwego doboru zdjęć lub trenowania sieci na innych typach danych niż te wykorzystane w eksperymencie. Dalsza analiza koncentrowała się na algorytmach o najwyższej skuteczności klasyfikacji, ponieważ pozostałe nie wносиły wartości badawczej.

5.6.1. Kompresja

Eksperymenty wykazały, że kompresja zdjęcia może wpływać na skuteczność klasyfikacji algorytmów. Przetestowano trzy poziomy kompresji – 50%, 20% i 10%. Wszystkie algorytmy odnotowały niewielki spadek skuteczności wraz ze wzrostem

poziomu kompresji, co było zgodne z oczekiwaniami, ponieważ wyższa kompresja prowadzi do utraty szczegółów obrazu. Najlepsze wyniki uzyskiwał FaceNet512, a najniższe VGG-Face. ArcFace osiągał najniższe spadki przy 50% i 20% kompresji, nie odnotowując spadku przy 50%, lecz przy 10% wykazał największy spadek spośród wszystkich algorytmów. Pozostałe algorytmy wykazywały stabilne spadki: do 0,5% przy 50%, do 1,5% przy 20% i do 3,2% przy 10%.

Tabela 5.7: Procentowy spadek precyzji modeli rozpoznawania twarzy przy różnych poziomach kompresji

Model	Kompresja 50%	Kompresja 20%	Kompresja 10%
ArcFace	0%	0,9%	4,1%
FaceNet	0,5%	1,3%	2,6%
FaceNet512	0,8%	1,4%	3,2%
VGG-Face	0,8%	1,4%	2,6%

Rozmiary baz danych znacznie się zmniejszyły po kompresji – od około 55% przy 50% kompresji do około 80% przy 10%. Różnica w rozmiarze między poziomami 20% i 10% była już już mniejsza. Biorąc pod uwagę skuteczność algorytmów i oszczędność miejsca, najlepszym kompromisem okazała się kompresja 20%.

Tabela 5.8: Rozmiary skompresowanych baz zdjęć i oryginalnej bazy

Normalne	Kompresja 50%	Kompresja 20%	Kompresja 10%
179 MB	79,7 MB	50,8 MB	37,5 MB

5.6.2. Szum

Zdjęcia zostały poddane działaniu szumów, aby sprawdzić ich wpływ na jakość klasyfikacji algorytmów. Dodanie szumu symuluje sytuacje, w których przetwarzane obrazy są zakłócone np. przez utratę sygnału lub jego osłabienie. Sprawdzone również możliwość odzyskania skuteczności poprzez zastosowanie filtracji medianowej.

5.6.2.1. Szum biały (Gaussa)

Wszystkie analizowane algorytmy odnotowały spadek skuteczności po dodaniu szumu Gaussa. Było to zgodne z oczekiwaniami, ponieważ zakłócony obraz traci część cech, które umożliwiają prawidłową klasyfikację.

Tabela 5.9: Procentowy spadek precyzji algorytmów po dodaniu szumu Gaussa

Model	Spadek precyzji po szumie Gaussa
ArcFace	20,38%
FaceNet	9,17%
FaceNet512	3,25%
VGG-Face	5,98%

Tabela 5.10: Procentowy spadek precyzji algorytmów po szumie S&P na poziomie 10%

Model	Spadek precyzji po szumie 10% S&P
ArcFace	42,58%
FaceNet	35,18%
FaceNet512	19,27%
VGG-Face	26,52%

Tabela 5.11: Procentowy spadek precyzji algorytmów po szumie S&P na poziomie 5%

Model	Spadek precyzji po szumie 5% S&P
ArcFace	20,81%
FaceNet	14,71%
FaceNet512	8,48%
VGG-Face	12,19%

5.6.3. Rozmycie

Zdjęcia rozmyte pionowo i poziomo z użyciem filtrów o rozmiarze 5 i 10 pikseli wykazały różny wpływ na skuteczność algorytmów. Rozmycie 5 pikseli wpływało nieznacznie na skuteczność, bez istotnej utraty jakości klasyfikacji. Natomiast rozmycie 10 pikseli powodowało zauważalny spadek skuteczności we wszystkich algorytmach.

W przypadku rozmycia poziomego przy 10 pikselach algorytmy utrzymywały zdolność klasyfikacyjną przy pewnym spadku dokładności – precyzja i czułość pozostawały względnie zrównoważone. Inaczej było w przypadku rozmycia pionowego – tu spadek skuteczności był większy. Większość algorytmów (poza FaceNet512) wykazywała znaczący spadek czułości i wzrost precyzji, co oznaczało większą tendencję do błędnej klasyfikacji zdjęć jako przedstawiających tę samą osobę.

Wyniki te podkreślają, że kierunek i intensywność rozmycia mają istotne znaczenie dla dokładności rozpoznawania twarzy. Rozmycie poziome częściowo zachowuje funkcjonalność przy spadku precyzji, natomiast rozmycie pionowe i większe rozmiary okna filtra zwiększają ryzyko błędnej klasyfikacji – wyjątkiem był tu model FaceNet512, który wykazał największą odporność na te zmiany.

5.7. Wnioski

Celem niniejszego badania było sprawdzenie skuteczności algorytmów rozpoznawania twarzy w zależności od poziomu zniekształcenia zdjęcia, co pozwoliło potwierdzić postawioną hipotezę badawczą. Zniekształcenia zastosowano na trzy sposoby: poprzez kompresję zdjęć, rozmycie oraz dodanie szumu.

Kompresja zdjęć miała najmniejszy wpływ na jakość klasyfikacji spośród trzech analizowanych zniekształceń. Pomimo zastosowania wysokiego poziomu kompresji – do 10% – algorytmy traciły niewiele skuteczności. Eksperymenty wykazały, że w celu oszczędności miejsca na dysku najlepiej zastosować kompresję zdjęć do poziomu 20%; utrata jakości klasyfikacji jest wtedy niewielka, a rozmiar bazy danych istotnie zmniejszony.

Zastosowanie szumu miało większy wpływ na jakość klasyfikacji niż poziom kompresji. Sprawdzone, jak zmienia się skuteczność algorytmów po dodaniu szumu Gaussa oraz szumu typu sól i pieprz (S&P) na dwóch poziomach – 5% i 10%, czyli zastąpieniu odpowiednio 5 i 10 procent pikseli losowymi wartościami. Spadek skuteczności klasyfikacji był porównywalny pomiędzy szumem Gaussa a szumem 5% S&P. ArcFace odnotował zdecydowanie największy spadek skuteczności. Zastosowanie szumu 10% S&P skutkowało znacznym spadkiem jakości klasyfikacji we wszystkich algorytmach. Po zastosowaniu filtra medianowego do zdjęć z szumem algorytmy odzyskały większość swojej skuteczności i ponownie prawidłowo klasyfikowały zdjęcia do odpowiednich klas.

W przypadku rozmycia istotne znaczenie miał kierunek rozmycia. Algorytmy dobrze radziły sobie z rozmyciem poziomym, natomiast w przypadku rozmycia pionowego algorytmy ArcFace i FaceNet stawały się bezużyteczne – klasyfikowały większość zdjęć jako zdjęcia tej samej osoby, niezależnie od danych wejściowych.

Algorytmem, który zdecydowanie najlepiej poradził sobie we wszystkich eksperymentach, był FaceNet512. Algorytmy OpenFace, DeepFace oraz DeepID uzyskały bardzo słabe wyniki na testowanych danych. Mogło to wynikać z niewłaściwego doboru danych lub błędnego przetrenowania modeli. ArcFace uzyskał wysoką skuteczność na oryginalnej bazie, jednak jego jakość klasyfikacji najbardziej spadła przy silnych zniekształceniach. Może to wynikać z zastosowanego w modelu algorytmu Additive Angular Margin Loss. Wykazano, że kompresja zdjęć może znacząco zmniejszyć rozmiar bazy danych, co jest istotne w przypadku ograniczonej przestrzeni dyskowej i dużej liczby zdjęć, bez powodowania istotnych strat w skuteczności algorytmów. Pokazano również, że pomimo znacznego spadku skuteczności przy zaszumieniu zdjęcia, możliwe jest odzyskanie skuteczności algorytmów poprzez zastosowanie odpowiednich filtrów. W przypadku rozmycia najważniejszym czynnikiem okazał się jego kierunek – rozmycie pionowe powodowało znacznie większy spadek skuteczności, obniżając znacząco miarę czułości, czyli to, jak wiele elementów z danej klasy zostało poprawnie rozpoznanych.