Uniwersytet im. Adama Mickiewicza w Poznaniu
Wydział Matematyki i Informatyki

Generowanie i walidacja syntetycznych zbiorów danych obrazów do trenowania modeli sztucznej inteligencji w wizji komputerowej

Generating and Validating Synthetic Image Datasets for Training AI Models in Computer Vision

Jacek Kałużny

Rozprawa doktorska z dziedziny nauk ścisłych i przyrodniczych w dyscyplinie Informatyka

Promotor:
prof. UAM dr hab. Krzysztof Dyczkowski
Promotor pomocniczy:
dr Wojciech Pałubicki

Poznań 2024

Przede wszystkim chciałbym serdecznie podziękować moim Rodzicom za ich wsparcie, cierpliwość i wiarę we mnie przez cały okres mojej edukacji. Bez Waszej pomocy i motywacji, ta praca nie mogłaby powstać.

Podziękowania

Szczególne podziękowania kieruję do prof. UAM dr. hab. Krzysztofa Dyczkowskiego oraz dr. Wojciecha Pałubickiego za ich nieocenione wsparcie naukowe, cierpliwość i zaangażowanie w moją pracę.

Chciałbym również wyrazić wdzięczność wobec prof. dr. Farhah Assaad-Gerbert oraz Yannika Schreckenberga za ich współpracę i wsparcie.

Specjalne podziękowania należą się także prof. dr. Dominikowi L. Michelsowi, prof. dr. Sörenowi Pirkowi oraz prof. dr. Bedrichowi Benesowi za ich cenne rady i inspiracje, które znacząco przyczyniły się do rozwoju mojej pracy.

Abstract

This thesis explores the generation and validation of synthetic image datasets for training models in computer vision. The core objective is to address the challenges of data scarcity and variability in training robust and accurate AI models. A multi-faceted approach was adopted, involving the development of novel techniques for synthetic data generation, the creation of realistic and diverse datasets, and the rigorous validation of these datasets through comprehensive experiments.

The initial phase of the research focuses on the reconstruction of botanical trees from single images, leveraging advanced machine learning algorithms. A method was introduced using radial bounding volumes and bi-modal growth models to accurately reconstruct 3D tree structures. This technique was validated through extensive testing against real-world datasets, demonstrating its efficacy in producing high-fidelity reconstructions.

Subsequently, the investigation was expanded to synthetic dataset creation, utilizing ControlNet integrated with Stable Diffusion to generate realistic synthetic images of various plants. This approach includes a detailed pipeline for generating annotated images, ensuring the datasets are suitable for training deep learning models for tasks.

The final phase involves the validation of the synthetic datasets. A series of experiments were conducted to compare the performance of AI models trained on synthetic data versus those trained on real data. The results indicate that models trained on these synthetic datasets perform comparably to, and in some cases exceed, those trained on traditional datasets. This highlights the potential of synthetic data to supplement or even replace real data in certain applications.

This research contributes to the field of computer vision by providing robust methods for synthetic data generation and validation, paving the way for more efficient and cost-effective training of AI models. The findings have significant implications for various applications, including agricultural automation, urban planning, and beyond.

Streszczenie

Niniejsza rozprawa doktorska dotyczy generowania i walidacji syntetycznych zbiorów danych obrazów do trenowania modeli w widzeniu komputerowym. Głównym celem pracy jest rozwiązanie problemów związanych z niedoborem danych oraz ich zmiennością w kontekście trenowania solidnych i dokładnych modeli SI. Zaproponowano wieloaspektowe podejście obejmujące opracowanie nowych technik generowania danych syntetycznych, tworzenie realistycznych i zróżnicowanych zbiorów danych oraz rygorystyczną walidację tych zbiorów poprzez kompleksowe eksperymenty.

Początkowa faza badań skupia się na rekonstrukcji botanicznych drzew na podstawie pojedynczych obrazów z wykorzystaniem zaawansowanych algorytmów uczenia maszynowego. Zaproponowana została metoda wykorzystująca "Radial Bounding Volumes" oraz dwumodalne modele wzrostu, aby dokładnie odtworzyć struktury 3D drzew. Technikę zwalidowano poprzez intensywne testy na rzeczywistych zbiorach danych, wykazując jej skuteczność w tworzeniu wysokiej jakości rekonstrukcji.

Badania zostały rozszerzone na tworzenie syntetycznych zbiorów danych poprzez wykorzystywanie ControlNet zintegrowanego z Stable Diffusion do generowania realistycznych syntetycznych obrazów roślin. To autorskie podejście obejmuje szczegółowy proces tworzenia anotowanych obrazów, zapewniając, że zbiory danych są odpowiednie do trenowania modeli głębokiego uczenia.

Ostatnia faza pracy obejmuje walidację syntetycznych zbiorów danych. Przeprowadzony został szereg eksperymentów porównujących wydajność modeli SI trenowanych na danych syntetycznych z tymi trenowanymi na danych rzeczywistych. Wyniki wskazują, że modele trenowane na powstałych syntetycznych zbiorach danych działają porównywalnie, a w niektórych przypadkach przewyższają modele trenowane na tradycyjnych zbiorach danych. To podkreśla potencjał danych syntetycznych do uzupełniania lub nawet zastępowania danych rzeczywistych w niektórych aplikacjach.

Badania te wnoszą wkład w dziedzinę widzenia komputerowego poprzez dostarczenie solidnych metod generowania i walidacji danych syntetycznych, torując drogę do bardziej efektywnego i ekonomicznego trenowania modeli SI. Wyniki mają istotne implikacje dla różnych zastosowań, w tym automatyzacji rolnictwa, badań biologicznych i innych.

Spis treści

Spis rysunków

Spis tabel

Rozdział 1

Wstęp

W ostatnich latach integracja uczenia maszynowego i analizy danych w różnych dziedzinach badawczych stała się coraz bardziej istotna. Zdolność do analizy złożonych i różnorodnych zbiorów danych, niezależnie od ich pochodzenia, niesie ze sobą znaczący potencjał do postępu w naszym rozumieniu różnych procesów oraz poprawy praktycznych zastosowań, takich jak rolnictwo, medycyna i zarządzanie środowiskiem [96, 80, 15]. Niemniej jednak analiza danych w tych kontekstach stawia wyjątkowe wyzwania w porównaniu z innymi rodzajami danych.

Dane, z którymi pracują badacze, są z natury bardziej zróżnicowane i złożone niż dane z wielu innych dziedzin. W przeciwieństwie do statycznych, dobrze zdefiniowanych zbiorów danych, mogą się one znacząco różnić pod względem struktury, skali i złożoności. Na przykład zbiory danych mogą obejmować różnorodne źródła, takie jak sekwencje genetyczne, struktury molekularne, dane obrazowe oraz pomiary ekologiczne, z których każdy ma swoje unikalne cechy i wymagania analityczne [17]. Ta różnorodność wymaga zaawansowanych narzędzi i modeli obliczeniowych zdolnych do obsługi niuansów informacji [21, 78].

Ponadto zmienność w danych, taka jak różnorodność morfologiczna lub plastyczność fenotypowa w odpowiedzi na zmiany środowiskowe, dodaje kolejny poziom złożoności. Modele uczenia maszynowego i algorytmy obliczeniowe muszą być odporne i elastyczne, aby dokładnie przetwarzać i interpretować takie dane. Modele te muszą być trenowane na obszernych, dobrze oznakowanych zbiorach danych, aby skutecznie uogólniać się w różnych kontekstach [1, 32].

Kluczowym czynnikiem sukcesu analizy danych jest sama ich jakość. Wysokiej jakości dane zapewniają, że modele mogą się uczyć i dokonywać trafnych prognoz. W moich badaniach koncentruję się na analizie danych z obrazów, zwłaszcza fotografii wykonanych telefonem. To podejście jest szczególnie istotne, ponieważ w terenie często nie ma dostępu do specjalistycznego sprzętu, ale niemal zawsze jest dostęp do smartfona. Wygoda i dostępność

użycia aparatów w telefonach mogą ułatwić zbieranie danych, czyniąc ten proces prostszym i tańszym dla badaczy w celu uchwycenia obrazów okazów do dalszej analizy [17].

Przedstawione badania mają na celu opracowanie metod do dokładnej analizy danych z pojedynczych fotografii wykonanych smartfonami. To podejście ma potencjał demokratyzacji zbierania danych, umożliwiając większej liczbie badaczy efektywne i skuteczne ich gromadzenie i analizowanie, niezależnie od dostępu do specjalistycznego sprzętu [17, 80]. Wykorzystując wszechobecność aparatów w smartfonach, możemy zwiększyć zdolność badaczy do prowadzenia badań w różnych środowiskach: od odległych lokalizacji terenowych po środowiska miejskie [21].

Pomimo potencjału, istnieje znaczne wyzwanie w pozyskaniu wysokiej jakości, dobrze oznakowanych danych. Tradycyjne metody zbierania i anotacji danych są pracochłonne i czasochłonne, szczególnie w przypadku badań szczegółowych. Aby temu sprostać, niedawne osiągnięcia skupiają się na generowaniu danych syntetycznych za pomocą modeli proceduralnych i technik generatywnej SI [1, 78]. Takie podejścia podkreślają przydatność danych syntetycznych w trenowaniu modeli głębokiego uczenia, zmniejszając zależność od ręcznie anotowanych zbiorów danych [1].

Proceduralne generowanie danych polega na tworzeniu danych na podstawie zdefiniowanych reguł, co zapewnia kontrolowaną zmienność i różnorodność. Przykładowo, Fadaeddini et al. (2018) [19] pokazali, że generatywne sieci neuronowe, takie jak GAN, mogą być stosowane do proceduralnego generowania tekstur w grach wideo, co zwiększa realizm i różnorodność generowanych danych. W szerszym kontekście proceduralne generowanie znajduje zastosowanie w takich dziedzinach, jak: inspekcja defektów, analiza obrazów medycznych, a także w wizji komputerowej, gdzie randomizacja domeny pozwala na lepsze trenowanie modeli [27, 28].

Modele dyfuzyjne, takie jak Stable Diffusion, otwierają nowe możliwości w generowaniu danych syntetycznych, zwłaszcza w kontekście segmentacji semantycznej i klasyfikacji obrazów. Nguyen et al. (2023) [56] pokazali, że wykorzystanie modeli dyfuzyjnych do generowania syntetycznych obrazów z maskami segmentacyjnymi na poziomie pikseli pozwala na skuteczne trenowanie modeli bez potrzeby ręcznego anotowania danych. Modele te mogą być także wykorzystywane w bardziej technicznych dziedzinach, takich jak generowanie złożonych scen 3D i analiza obrazów w medycynie, co pokazuje ich szerokie zastosowanie [82].

Niniejsza rozprawa doktorancka zajmuje się problemem zbierania i tworzenia danych, koncentrując się na zastosowaniu sztucznych sieci neuronowych i widzenia komputerowego do zapewnienia prostych, niezawodnych rozwiązań. Pokazuje wpływ danych syntetycznych,

które są kluczowe dla problemów, nie generujących wystarczającej ilości rzeczywistych punktów danych.

Główne hipotezy stosowane w niniejszej dysertacji:

1. Sztuczne sieci neuronowe mogą być wykorzystywane do zbierania, wyodrębniania i analizy danych.

2. Modele proceduralne generujące zbiory danych syntetycznych są rozwiązaniem problemu małych zbiorów danych.

3. Integracja danych syntetycznych z rzeczywistymi zbiorami danych zwiększa odporność i uogólnialność modeli uczenia maszynowego.

Rozdział 2 koncentruje się na rekonstrukcji geometrii pojedynczego drzewa za pomocą zaledwie jednej fotografii. Metody rekonstrukcji modeli 3D z danych czujników często napotykają wyzwania, takie jak zacieranie struktur przez liście oraz złożoność wzorców rozgałęzień [1].

Tradycyjne podejścia, czyli użycie skanerów laserowych lub wielu obrazów, oferują wysoką dokładność, ale nie zawsze są wykonalne dla zastosowań na dużą skalę. Prace, autorstwa Neubert et al. (2007) [53] i Livny et al. (2011) [47] pokazały zastosowanie segmentacji obrazu i chmur punktów do rozwiązania niektórych z tych wyzwań. Jednak generowanie realistycznych modeli 3D z pojedynczych obrazów pozostaje otwartym problemem, często wymagającym innowacyjnych rozwiązań, takich jak integracja modeli proceduralnych i sieci neuronowych w celu zniwelowania luki między danymi syntetycznymi a rzeczywistymi [80]. Bardziej niedawne osiągnięcia można zaobserwować w polu neuroradiacyjnych pól (NeRF), które pokazują obiecujący potencjał w syntezowaniu nowych widoków złożonych scen poprzez optymalizację podstawowej ciągłej funkcji sceny wolumetrycznej przy użyciu rzadkiego zestawu widoków wejściowych. Pomimo ich skuteczności, techniki NeRF nadal zazwyczaj wymagają wielu fotografii o znanych pozycjach kamer, aby wygenerować wysokiej jakości rekonstrukcje, co podkreśla złożoność rekonstrukcji 3D z pojedynczego obrazu [51].

Po problemie zajmującym się geometrią całego drzewa warto się skupić na jego części. Umożliwi to dokładniejsze pomiary oraz większe dopasowanie do konkretnych potrzeb i zastosowań.

Rozdział 3 omawia utworzenie zbioru danych LAESI (Leaf Area Estimation with Synthetic Imagery), który stanowi znaczący zasób do analizy morfologii liści poprzez generowanie syntetycznych obrazów liści z precyzyjnymi anotacjami przy użyciu modeli proceduralnych i wypełniania opartego na ControlNet [96]. Ten rozdział szczegółowo opisuje

nie tylko tworzenie, ale też zastosowanie zbioru danych LAESI w trenowaniu modeli uczenia maszynowego do szacowania powierzchni liści, podkreślając korzyści danych syntetycznych w badaniach botanicznych.

Rozdział 4 kontynuuje tworzenie zbiorów danych z wykorzystaniem połączenia modelu proceduralnego w połączeniu z ControlNet. Tym razem jednak jest brane pod uwagę nawet kilka roślin w specjalnej doniczce do kontrolowania klimatu (lizymetrze [24]). Proces generowania danych jest rozbudowany w porównaniu do procesu z poprzedniego rozdziału. Dochodzą nowe elementy, takie jak wykorzystanie map głębokości, czy techniki takie jak Low-Rank Adaptation (LoRA) [36]. Wszystko to w celu poprawienia poprawności otrzymanych danych. Przybliża to też niniejsze dane do celu działania tej sieci - stworzenia mapy głębokości.

Rozdział 2

Rekonstrukcja geometrii na podstawie pojedynczego zdjęcia

2.1 Wprowadzenie

Ze względu na dominującą obecność roślinności nie tylko w środowiskach zewnętrznych, ale także w miejskich i nawet wewnętrznych przestrzeniach, szczegółowe modele roślin znacznie przyczyniają się do realizmu niemal wszystkich wirtualnych scen. Zakres zastosowań obejmuje architekturę i rolnictwo, badania leśne i planowanie urbanistyczne, szkolenie autonomicznych agentów, zrozumienie scen dla rzeczywistości rozszerzonej oraz tworzenie treści do gier i filmów. Ważne jest, aby rekonstruować modele tak wiernie, jak to możliwe, aby uchwycić cechy obserwowanych roślin, szczególnie w kontekstach, w których konieczne jest podejmowanie świadomych decyzji dotyczących drzew lub interakcja z roślinnością.

Aktualne metody rekonstrukcji drzew na podstawie danych z czujników opierają się głównie na osobnej rekonstrukcji głównej struktury rozgałęzień i korony drzewa. Centralnym elementem wielu technik jest uzyskanie geometrycznych obwiedni dla korony drzewa, które są następnie wypełniane gałęziami generowanymi przez model proceduralny. W przypadku obrazów można użyć szkiców zdefiniowanych przez użytkownika [85] lub segmentacji obrazów [4], aby wnioskować o takich obwiedniach 3D. Neubert et al. [54] używają dwóch obrazów do uzyskania reprezentacji voxelowej 3D, aby ograniczyć niejednoznaczność widoku i następnie modelować struktury rozgałęzień za pomocą algorytmu przepływu cząsteczek. W przypadku chmur punktów obwiednie 3D można generować łatwiej i w bardziej subtelny sposób przed modelowaniem geometrii gałęzi [46].

Rekonstrukcja drzew jest trudna z wielu powodów: po pierwsze, listowie często zasłania części głównej struktury rozgałęzień, co utrudnia ich rekonstrukcję. Po drugie, nawet jeśli

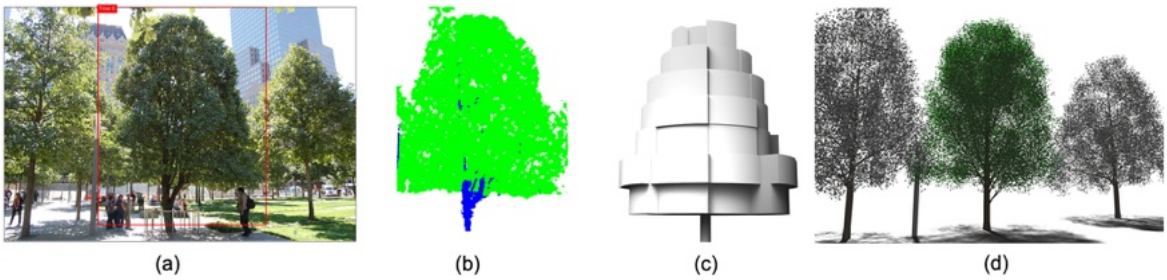

Fig. 2.1 Rekonstrukcja drzewa z pojedynczego obrazu: metoda automatycznie rekonstruuje drzewa z pojedynczych obrazów (a). Maski segmentacji semantycznej drzew są uzyskiwane, aby zidentyfikować gałęzie i liście (b). Wprowadzane jest Radial Bounding Volumes (c) jako reprezentacja o stałym rozmiarze dla modeli drzew 3D. Ta reprezentacja może być bezpośrednio uczona za pomocą sieci neuronowych. Przewidziane RBV są następnie używane do automatycznej rekonstrukcji modeli drzew z wysokim stopniem wierności wizualnej (d). Aby zrekonstruować wiele drzew na jednym obrazie, wykrywane są ramki ograniczające (a, czerwone) przed uzyskaniem masek segmentacji semantycznej.

gałęzie są widoczne, ich struktura jest skomplikowana i nie zawsze może być w pełni uchwycona przez dzisiejszy sprzęt sensoryczny, taki jak kamery czy skanery laserowe. Po trzecie, rekonstrukcja drzewa z jednego punktu widzenia jest zadaniem niejednnoznacznym, ponieważ istnieje wiele możliwych rozwiązań modelowania odpowiedniej geometrii drzewa 3D. Jednak często pożądane jest zrekonstruowanie drzewa z tylko jednego punktu widzenia, mimo tych wyzwań. Nie zawsze możliwe jest uzyskanie dostępu do drzewa z wielu kierunków. Co więcej, rekonstrukcja drzew przy użyciu popularnych urządzeń sensorycznych, takich jak kamery w telefonach lub dronach, jest korzystna w porównaniu do bardziej precyzyjnego, lecz mniej wygodnego sprzętu, takiego jak skanery laserowe.

W przypadku wielu zastosowań, takich jak rekonstrukcja środowisk miejskich, konieczne jest generowanie dużych ilości modeli drzew 3D, aby realistycznie wypełnić sceny. Jednak żadne z istniejących metod rekonstrukcji nie pozwalają na rekonstrukcję drzew z pojedynczych obrazów na dużą skalę, ponieważ wymagają one albo adnotacji dostarczonych przez użytkownika dla każdego drzewa, albo czasochłonnej i nieintuicyjnej regulacji parametrów modeli proceduralnych.

Aby sprostać temu otwartemu problemowi badawczemu, potrzebne jest nowe podejście do rekonstrukcji rzeczywistych drzew z pojedynczych fotografii. Opracowana metodologia opiera się na najnowocześniejszych sieciach neuronowych, które pozwalają maskować drzewa na zdjęciach, identyfikować ich gatunki i uczyć się ich ogólnej struktury geometrycznej jako 3D Radial Bounding Volumes. Wykorzystano uzyskane cechy wraz z modelem proceduralnym do generowania modelu drzewa, który uchwyca ogólny

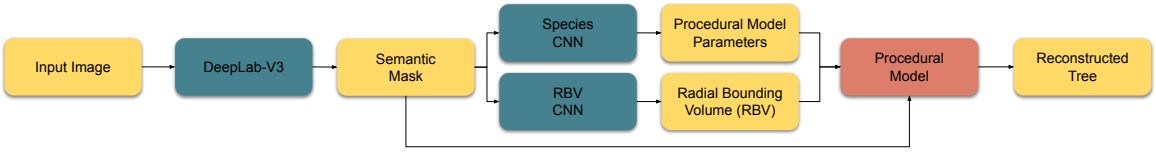

Fig. 2.2 Ogólny schemat działania: wykorzystywany jest pojedynczy obraz jako dane wejściowe i stosowany jest DeepLab-V3 do segmentacji semantycznej, aby uzyskać maskę semantyczną obrazu, która przypisuje każdemu pikselowi jedną z trzech etykiet: tło, gałęzie oraz listowie. Maska segmentacji służy do trenowania sieci neuronowej do rozpoznawania gatunków oraz innej sieci neuronowej do szacowania radialnych woluminów ograniczających (RBV). Najpierw drzewa są uzyskiwane na podstawie przewidywanego gatunku i wartości parametrów dla modelu rozwojowego. Następnie przewidywana maska semantyczna, oszacowane RBV, oraz wybrane wartości parametrów gatunków są wykorzystywane do obliczenia wzrostu modelu drzewa 3D z cechami wizualnymi odpowiadającymi obrazowi wejściowemu. Kolory wyróżniają dane wejściowe, wyjściowe oraz generowane (żółty), użyte sieci neuronowe (morski) oraz model proceduralny (czerwony).

wygląd uchwyconego drzewa. Zostało pokazane, że wyuczone cechy umożliwiają dostrojenie modelu proceduralnego, aby uchwycić subtelne niuanse formy drzewa w różnych gatunkach. 3D Radial Bounding Volume, wraz z ograniczeniami morfologicznymi modelu proceduralnego, umożliwia wierne generowanie geometrii drzew, która nie jest widoczna na zdjęciach wejściowych. Ponadto, zostało pokazane, że opracowane w ramach badań sieci neuronowe można trenować na wyłącznie obrazach syntetycznie generowanych drzew, aby przewidzieć parametry dla rzeczywistych drzew pokazanych na fotografiach. Aby dodatkowo zaprezentować przydatność niniejszej metody do tworzenia treści, wykorzystano podejście oparte na fizyce do symulacji i animacji ruchu drzew. Gdy model drzewa jest wygenerowany, jego graf jest przekształcany w reprezentację dynamiki opartej o pozycje i jest realistycznie symulowany ruch roślin oparty na modelu Cosserat rods[61].

Przykład na rysunku 2.1 ilustruje możliwości opracowanej metody. Najpierw obliczana jest maska segmentacji semantycznej, która przypisuje etykiety klas do gałęzi i liści. Następnie, na podstawie tej maski generowania jest trójwymiarowa strukturę o stałej rozdzielczości przestrzennej, którą zostaje nazwana Radial Bounding Volume. Maska i ta obwiednia są wykorzystywane do kontrolowania wzrostu modelu rozwojowego drzewa w celu wygenerowania modelu 3D.

Rekonstrukcje drzew z rzeczywistych fotografii przedstawiono na rysunku 2.14.

2.2 Przegląd literatury

Badania nad modelowaniem drzew i roślin cieszą się znacznym zainteresowaniem. Wczesne podejścia do modelowania struktur rozgałęzień używają algorytmów opartych na regułach

[33, 65], powtarzalnych wzorców [39, 3, 58, 81], automatów komórkowych [25], systemów cząsteczek [72], lub kombinacji tych podejść [45]. Ponieważ rośliny rzadko rosną w izolacji, wiele metod koncentruje się na interakcji roślin ze środowiskiem za pomocą modułów zapytaniowych [52] lub algorytmów losowych [7], odwróconego modelowania proceduralnego [83, 26], poprzez modelowanie konkurencji o zasoby i samoorganizację [59], kolonizację przestrzeni [76], lub modelowanie reakcji wzrostu [64].

Ostatnio szereg metod zaczęło koncentrować się na dokładniejszym modelowaniu biomechanicznych i fizycznych właściwości roślin [60]. Zakres tych badań obejmuje modelowanie interaktywnego wzrostu drzew [48, 62] i pnączy [30], interakcji roślin i płynów, takich jak wiatr [63, 29, 67], oraz ognia [61], aż po symulację właściwości materiałowych za pomocą metod elementów skończonych (FEM) [99, 88], a nawet kambialnego wzrostu drzew [41]. Metody te dostarczają modeli obliczeniowych i reprezentacji, które umożliwiają dynamiczne ruchy geometrii roślin, zwiększając realizm.

Innym sposobem są metody oparte na szkicowaniu, które mają na celu realistyczne modelowanie struktur rozgałęzień, zachowując artystyczną kontrolę. Większość metod polega na szkicach zdefiniowanych przez użytkownika, aby modelować kluczowe cechy drzew [57, 85], roślin [2], a nawet kwiatów [37]. Szkice definiowane przez użytkownika służą jako potężne narzędzie umożliwiające generowanie złożonych geometrii drzew. Przykłady obejmują konwersję szkiców odręcznych poprzez optymalizację probabilistyczną [13], użycie szkiców gałęzi do kierowania przepływami cząsteczek w celu rekonstruowania struktur drzew [54], czy definiowanie kształtów obwiedni na podstawie szkicowanych sylwetek [92].

Zamiast ręcznego modelowania roślin, uznano, że rekonstrukcja realistycznych struktur rozgałęzień bezpośrednio z danych sensorowych jest intrygującą alternatywą. Szereg metod ma na celu rekonstrukcję modeli drzew z różnych źródeł danych, w tym obrazów [71, 54, 66, 86], chmur punktów [93, 46], a nawet wideo [43]. Ze względu na ogromną złożoność geometryczną roślinności, celem rekonstrukcji modeli drzew i roślin, wiernych wszystkim ich cechom definiującym, pozostaje trudny i otwarty problem [16, 6, 55]. Wiele metod opiera się na danych z chmur punktów i dekompozycji procesu rekonstrukcji na wiele kroków lub komponentów. Na przykład Xu et al., Livny et al. [93, 46] wyraźnie rekonstruują główne gałęzie za pomocą algorytmu grafowego, podczas gdy listowie jest rekonstruowane jedynie w przybliżeniu na podstawie zestawu obwiedni. Bradley et al. [9], z drugiej strony, wykorzystują chmury punktów uzyskane z obrazów stereoskopowych i uczą model statystyczny połączony z dopasowaniem siatek nieregularnych w celu rekonstrukcji gęstego listowia. Li et al. [44] idą jeszcze dalej i uwzględniają domenę czasową do rekonstrukcji roślin, śledząc wydarzenia topologiczne, takie jak pączkowanie i bifurkacja.

W porównaniu do chmur punktów, obrazy stanowią wygodniejszy sposób uchwycenia roślinności, ponieważ kamery są łatwo dostępne w różnych urządzeniach, takich jak telefony czy drony. Jednak rekonstrukcja drzew z obrazów z pojedynczego widoku jest trudna. Jeden obraz nie dostarcza wystarczających szczegółów, aby znacząco zrekonstruować strukturę 3D drzewa. Szczegóły, które nie zostały uchwycone na obrazie, nie mogą być łatwo zrekonstruowane. Aby rozwiązać ten problem, Reche-Martinez et al. [71] i Neubert et al. [54] używają obrazów z wielu widoków, aby uzyskać bardziej holistyczne reprezentacje 3D drzew, co pozwala im na dokładniejsze zrekonstruowanie struktury rozgałęzień. Quan et al. [66] i Tan et al. [86] rejestrują sekwencję obrazów i używają techniki *structure from motion* do wyodrębnienia chmur punktów 3D, a następnie zrekonstruowania szczegółowych modeli drzew i roślin. Argudo et al. [4] używają segmentacji obrazów do generowania siatek obwiedni 3D koron drzew wraz z mapami odległości promieniowych, aby renderować rekonstrukcje drzew na skalę terenu.

Najbliżej tej pracy jest metoda Tan et al. [85], która również koncentruje się na rekonstrukcji drzew z pojedynczych obrazów. W tej pracy użytkownik musi ręcznie zidentyfikować główne gałęzie i kształt korony, rysując ogólny szkic. Geometria korony jest następnie generowana z predefiniowanych szablonów gałęzi, a liście są syntezowane na podstawie wygenerowanej struktury rozgałęzień. W przeciwieństwie do ich metody, proponowany jest automatyczny mechanizm przetwarzania do rekonstrukcji drzew, który nie wymaga interwencji użytkownika.

2.3 Przegląd metody

Celem tej metody jest rekonstrukcja realistycznych modeli drzew 3D z pojedynczych obrazów w sposób efektywny i na dużą skalę, co jest wyzwaniem. Dany obraz drzewa może prowadzić do wielu możliwych rozwiązań 3D, które spełniają projekcję 2D. Przedstawione podejście oparte jest na uczeniu maszynowym i bazuje na nowej reprezentacji modeli drzew i dwuetapowym procesie rekonstrukcji, który podsumowano na rys. 2.2. Wprowadza *Radial Bounding Volumes (Radialne Objętości Ograniczające, RBV)* dla modeli drzew, które kodują formę drzewa jako zestaw cylindrycznych warstw sektorów. Każdy sektor przechowuje najbardziej wysunięty w przestrzeni zewnętrznej zakres struktury rozgałęzień. Zostało pokazane, że ta forma nie tylko służy jako sensowna reprezentacja do uchwycenia złożonej formy drzewa, ale także może reprezentować różne gatunki przy stałej liczbie warstw i sektorów. Ponadto RBV można oszacować na podstawie obrazów za pomocą uczenia głębokiego. Aby zrekonstruować drzewa z fotografii, najpierw przeprowadzona jest analiza obrazu oparta na sieci neuronowej do segmentacji semantycznej, która oddziela piksele

drzewa (liście i strukturę rozgałęzień) od innych pikseli (tło, inne obiekty). Sieć identyfikacji gatunku jest używana do wyodrębnienia gatunku drzewa obserwowanego na fotografii. Jest to wykorzystywane do dostrojenia modelu rozwoju drzew. Następnie wykorzystywana jest trzecia sieć neuronowa do przewidywania RBV drzewa na podstawie wywnioskowanej maski segmentacji semantycznej.

Po uzyskaniu RBV drzewa proces wzrostu modelu rozwoju drzew jest kierowany przy użyciu RBV i maski semantycznej. Rozwój roślin jest modelowany, łącząc podejście kolonizacji przestrzeni [76] z fenomenologicznym modelem wzrostu struktur rozgałęzień, co zostaje nazwane *bi-modal growth model*. Podczas gdy kolonizacja przestrzeni zapewnia sposób na kierowanie rozwojem gałęzi na podstawie punktów przyciągających rozproszonych w obwiedni 3D, fenomenologiczny wzrost generuje realistyczne i biologicznie prawdopodobne struktury rozgałęzień na podstawie liczby parametrów, które można uzyskać, identyfikator gatunku jest przewidywany za pomocą sieci identyfikacji gatunku. RBV i maska semantyczna uchwyconego drzewa są następnie używane do kierowania rozmieszczeniem markerów wzdłuż głównej struktury rozgałęzień i w przestrzeni 3D Radial Bounding Volume. Dzięki temu zestawowi można efektywnie i masowo rekonstruować modele drzew, które wykazują kluczowe cechy wizualne uchwyconych rzeczywistych drzew.

2.4 Rekonstrukcja drzewa z pojedynczego obrazu

W tej sekcji wprowadzone zostało pojęcie Radial Bounding Volumes (RBV) dla modeli drzew oraz zaproponowane zostały sieci neuronowe do segmentacji semantycznej, identyfikacji gatunku i szacowania RBV.

2.4.1 Radial Bounding Volumes

Koncept RBV (patrz rys. 2.3) wprowadzony został w celu zakodowania formy drzewa w sposób abstrakcyjny, który zachowuje kluczowe cechy kształtu. RBV jest generowany z pionowo zorientowanego cylindra o wysokości h i promieniu r zdefiniowanym przez maksymalną wysokość i maksymalny poziomy zasięg drzewa. Cylinder dzielę na n warstw L_i, gdzie każda warstwa reprezentuje cylindryczny plaster o rozmiarze h/n. Każda warstwa L_i jest następnie podzielona na stałą liczbę m sektorów S_{ij}. Promień każdego sektora zmienia się w zależności od ilości lokalnej zajętości. Każdy cylindryczny sektor jest dopasowany do geometrii drzewa w tej konkretnej lokalizacji, co skutkuje gęstą wklęsłą obwiednią.

W porównaniu do powszechnych reprezentacji modeli drzew, takich jak grafy czy siatki, RBV oferują przewagę w postaci kodowania formy drzewa na podstawie stałej liczby

sektorów. W porównaniu do innych reprezentacji o stałym rozmiarze, takich jak voxele, RBV kodują formę drzewa bardziej efektywnie, przechowując jedynie zewnętrzny kształt modelu drzewa.

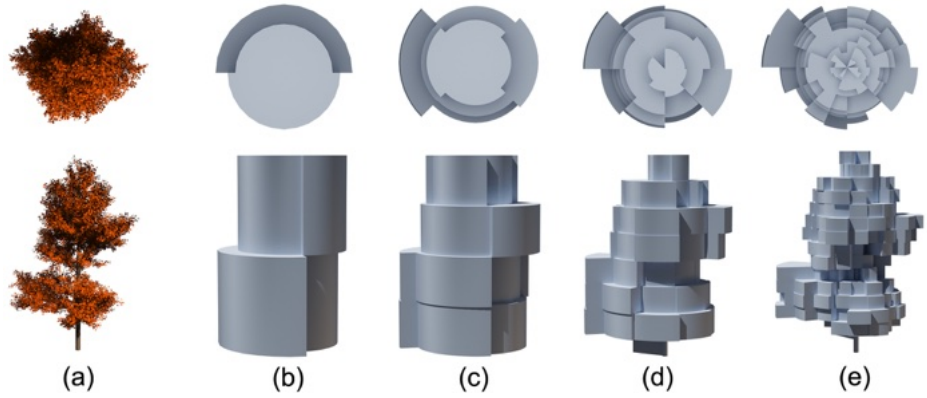

Fig. 2.3 Radialne Objętości Ograniczające drzewa klonowego (a) o rozdzielczościach 2×2 (b), 4×4 (c), 8×8 (d) i 16×16 (e).

Fig. 2.4 Przykłady różnych operacji augmentacji danych, w tym zmiany kontrastu, jasności i odcienia, a także transformacje takie jak: odwrócenie poziome, losowe przycięcie i rozmycie Gaussa.

2.4.2 Uczenie modeli rekonstrukcji drzew

Aby zrekonstruować modele drzew, przedstawiana metoda opiera się na trzech sieciach neuronowych, które pozwalają uzyskać maski listowia i struktury rozgałęzień na podstawie segmentacji semantycznej, identyfikować gatunki drzew oraz szacować 3D RBV. W dalszej części opisano architektury sieci i konfiguracje treningowe dla każdej z tych sieci.

2.5

Fig. 2.5 Przykład fotorealistycznego obrazu drzewa wygenerowanego w Unity przy użyciu Universal Render Pipeline.

Zbiór danych syntetycznych

Nie istnieją bazy danych rzeczywistych modeli drzew 3D. Dlatego, sieci neuronowe są trenowane wyłącznie w oparciu o syntetycznie generowane dane. Konkretnie, generowany jest zbiór danych z siedmioma różnymi gatunkami drzew (akacja, klon, dąb, jabłoń, sosna, wierzba, brzoza). To umożliwia wspólne zbieranie danych obrazowych, takich jak obrazy RGB renderowanych drzew I, maski semantyczne struktury rozgałęzień i listowia M, a także dane geometryczne, takie jak RBV R i siatka powierzchniowa O modelu drzewa oraz graf struktury rozgałęzień G. Przechowywane są również wartości parametrów P modelu rozwoju, które definiują gatunek wygenerowanego modelu drzewa, wraz z identyfikatorem gatunku U. Każdy punkt danych w zbiorze danych jest opisany przez krotkę $S = (I, M, R, O, G, P, U)$.

Aby trenować sieci neuronowe na syntetycznych danych i sprawić, by generalizowały na rzeczywiste fotografie, moim celem jest wygenerowanie fotorealistycznych obrazów drzew. Aby efektywnie generować duży zbiór danych, użyto programu napisanego w Unity z wykorzystaniem języka C# i Universal Render Pipeline (Fig. 2.5). Dodatkowo został stworzony model, który wykorzyste potok rastrowania oparty na OpenGL. Konkretnie, stosowany jest model PBR oparty na oświetleniu Blinn-Phong z silnika Unreal 4 [38].

Wiele źródeł światła o losowych kierunkach symuluje zróżnicowanie oświetlenia, które można zaobserwować na rzeczywistych fotografiach. Dodatkowo, stosowane są różne zestawy tekstur albedo, normalnych i szorstkości dla materiału powierzchniowego. Sektory RBV są definiowane za pomocą kanonicznej pozycji kamery. Generowane są obrazy RGB,

poprzez renderowanie każdego drzewo z losową rotacją na tle losowo wybranego krajobrazu lub sceny miejskiej. Maski semantyczne to obrazy RGB przechowujące trzy różne wartości kolorów dla tła (biały), gałęzi (niebieski) i listowia (zielony) i są bezpośrednio uzyskiwane z renderera. Zbiór danych składa się z 21 tysięcy pojedynczych krotek danych treningowych, 2,7 tysiąca krotek danych walidacyjnych i 2,7 tysiąca krotek danych testowych. Przykłady modeli drzew i masek są pokazane na rys. 2.4 i 2.6.

Semantyczna Segmentacja

Aby niezawodnie wykrywać maski instancji roślin wraz z podziałem na strukturę rozgałęzień i listowie, wykrywanie działa w oparciu o sieć DeepLab-V3, będącej jedną z najnowocześniejszych sieci do segmentacji semantycznej [12]. Architektura tej sieci wykorzystuje konwolucje atrous stosowane równolegle lub kaskadowo, aby uchwycić cechy obrazu na wielu skalach. Dzięki temu DeepLab-V3 skutecznie wyodrębnia maski semantyczne modeli drzew. Celem jest wykrywanie masek głównej struktury rozgałęzień oraz ogólnego kształtu listowia drzewa, aby wiernie rekonstruować drzewa z fotografii. Jednakże, trenowanie sieci wymaga danych referencyjnych dla masek semantycznych — precyzyjnych pikselowo etykiet różnych klas semantycznych — które są trudne do uzyskania. Na ile wiadomo, nie istnieje zbiór danych, który dostarczałby takich etykiet dla drzew. Dlatego użyte są rendery modeli drzew do wygenerowania syntetycznego zbioru danych.

Kiedy sieć neuronowa jest trenowana na danych syntetycznych, a powinna działać na rzeczywistych danych, to rozkłady danych dla fotografii rzeczywistych drzew i renderów modeli drzew często się nie pokrywają. Nazywa się to *luką domenową* i jest spowodowane artefaktami modelowania i renderowania — renderingi drzew nie mogą być efektywnie generowane z jakością zbliżoną do fotografii. Aby rozwiązać problem luki domenowej pomiędzy rozkładami danych, użyto strategii augmentacji danych opartej na zmianach konfiguracji renderingu i stosowaniu wielu transformacji obrazu. Podczas renderowania kamera jest ustawiana w losowej pozycji wokół drzewa, dostosowana jest liczba i pozycje świateł (do ośmiu) umieszczonych w scenie, a także zmieniana jest intensywność cieni. Następnie nakładane są transformacje obrazu na obrazie wejściowym, takie jak zmiany koloru, jasności i kontrastu. Dodatkowo, różnorodność jest rozszerzona o losową ilość rozmycia Gaussa, stosowanie losowego odwrócenia w poziomie oraz losowe kadrowanie obrazu dobranym oknem o minimalnym rozmiarze 256×256 pikseli.

Sieć DeepLab-V3 jest następnie trenowana na tych danych z użyciem standardowych ustawień do momentu jej zbieżności. Rys. 2.4 pokazuje przykłady renderowanych i augmentowanych obrazów drzew, a rys. 2.6 pokazuje maski semantyczne drzew uzyskane na podstawie renderowanych i prognozowanych danych. Należy zauważyć, że chociaż

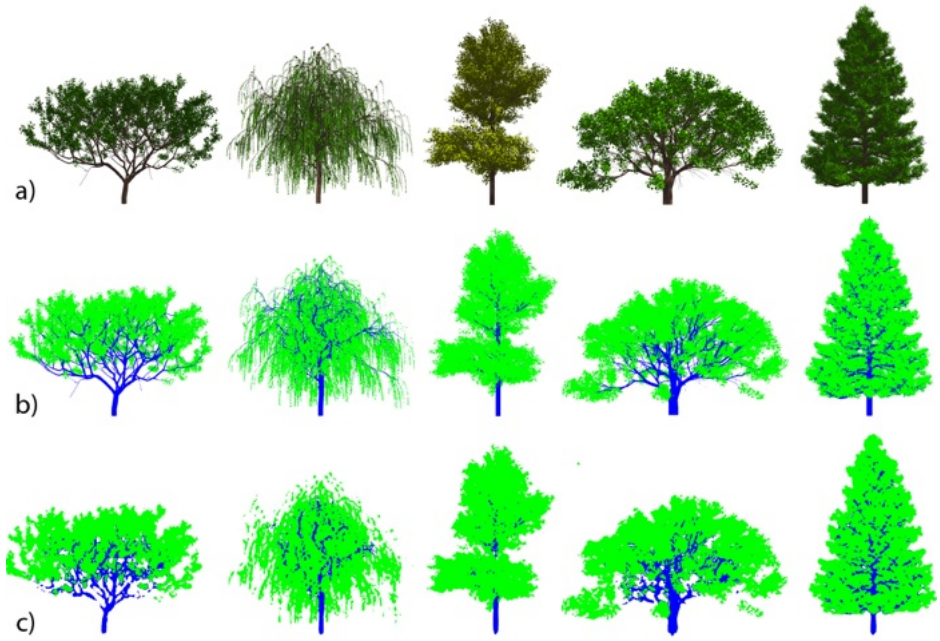

Fig. 2.6 Modele drzew i maski: Autorski program renderuje modele drzew (zapiswane jako obrazy RGB) (a) oraz generuje maski segmentacji semantycznej (b), które przypisują każdemu pikselowi jedną z trzech etykiet: tło, gałęzie i listowie. Następnie trenowana jest sieć segmentacji semantycznej na syntetycznych danych, aby oszacować maski segmentacji (c).

istnieje różnica w ilości szczegółów między wygenerowanymi a prognozowanymi maskami semantycznymi, nie jest wymagana idealna rekonstrukcji maski. Zamiast tego, celem jest wyodrębnienie głównej struktury rozgałęzień i ogólnego kształtu listowia, co łącznie dostarcza wystarczających informacji o drzewie, aby wiernie je zrekonstruować za pomocą algorytmu modelowania.

Uczenie modelu typu Radial Bounding Volumes

Ponieważ RBV kodują drzewa z ustaloną liczbą warstw i sektorów, mogą być bezpośrednio używane jako reprezentacja do trenowania sieci neuronowych. Konkretnie, trzeba opracować sposób na mapowanie maski semantycznej drzewa na jego RBV. Sieć do rozwiązania tego zadania można zdefiniować jako:

$$f_{RBV}(M) : \mathscr{M} \rightarrow \mathscr{R}, \tag{2.1}$$

gdzie $M \in \mathscr{M}$ oznacza maskę semantyczną, a $\mathscr{R}$ oznacza RBV. Celem jest oszacowanie wartości RBV za pomocą sieci neuronowej trenowanej do rozwiązania problemu regresji. Ponieważ regresja większej liczby parametrów jest zazwyczaj podatna na błędy, trenowane jest jednocześnie pięć wyjść, aby przewidzieć wartości RBV o różnych rozdzielczościach

(rys. 2.3). Regresja wartości dla RBV o niskiej rozdzielczości pomaga nadzorować regresję wartości dla RBV o wyższej rozdzielczości. Dlatego RBV jest definiowane z maksymalnie pięcioma poziomami rozdzielczości.

Na pierwszym poziomie definiuje się RBV jako jedną warstwę z jednym sektorem (tj. cylinder). Następnie stopniowo dzieli się tę objętość, aby wygenerować RBV o: 2 warstwach i 2 sektorach (poziom 2, 4 wartości), 4 warstwach i 4 sektorach (poziom 3, 16 wartości), 8 warstwach i 8 sektorach (poziom 4, 64 wartości) oraz 16 warstwach i 16 sektorach (poziom 5, 256 wartości). Aby trenować sieć jako kaskadę wyjść, łączy się wyjście z głowicy przewidującej zgrubszą rozdzielczość z osadzeniem obrazu i używa jako wejścia dla głowicy o wyższej rozdzielczości. Po zakończeniu treningu można wspólnie uzyskać wartości dla każdej z rozdzielczości RBV.

Sieć neuronowa f_{RBV}, trenowana do szacowania RBV jako kaskady czterech głowic, została przedstawiona na Rys. 2.7a. Wejściem do sieci są maski semantyczne, które są dzielone na trzy warstwy reprezentujące każdą z klas: tło, gałęzie oraz listowie. Wyjściowe wartości sektorów RBV, które wykorzystane są jako etykiety do trenowania sieci CNN dla RBV, są normalizowane dla każdego gatunku – współczynnik skalowania do uzyskania ostatecznego rozmiaru drzewa jest przechowywany w zestawie parametrów gatunku P. Maska jest osadzana za pomocą lekkiej architektury ConvNet (podobnej do AlexNet [42]) w celu wyodrębnienia cech obrazu (Rys. 2.7b). Konkretnie, wykorzystano 7 warstw konwolucji i operacji próbkowania w dół, aby uzyskać wektor osadzenia obrazu o szerokości 512. Następnie dodano trzy w pełni połączone warstwy, aby zdefiniować ostateczny wektor osadzenia, który wykorzystano jako wejście dla każdej z czterech głowic dekodujących. Każda głowica jest trenowana do rekonstrukcji wartości RBV dla konkretnej rozdzielczości.

Identyfikacja gatunków

Istotne jest również uzyskanie informacji o gatunku drzewa, aby w pełni automatycznie zrekonstruować modele drzew z fotografii. Celem jest nauczenie się mapowania od maski semantycznej drzewa do identyfikatora gatunku U. Sieć neuronową do tego zadania można zdefiniować jako:

$$f_{Gatunki}(M) : \mathscr{M} \rightarrow \mathscr{U} \ ,$$

gdzie $M \in \mathscr{M}$ oznacza maskę semantyczną, a $U \in \mathscr{U}$ oznacza identyfikator gatunku. Ponownie implementacja tej sieci jest jako architektura ConvNet, która umożliwia wyodrębnienie cech obrazu z masek semantycznych (rys. 2.7b). Na bazie wektora cech obrazu trenowane są trzy w pełni połączone warstwy z warstwą wyjściową do klasyfikacji (rys. 2.7c). Po uzyskaniu identyfikatora gatunku zostaje on wykorzystany do wybrania

zestawu zdefiniowanych parametrów P, które mogą być użyte z modelem rozwoju drzew do generowania drzew.

2.4.3 Rozwój drzew w trybie bi-modalnym

Wykorzystany model rozwoju zapewnia dwa tryby wzrostu. Po pierwsze, wyraża rozwój drzewa za pomocą fenomenologicznego modelu wzrostu [52, 83]. Dodatkowo, może wyrażać rozwój drzewa jako proces samoorganizacji gałęzi w przestrzeni [76, 59]. Podczas gdy fenomenologiczny tryb wzrostu symuluje rozwój drzewa zgodnie z obserwacjami z botaniki na poziomie gałęzi, tryb samoorganizacji jest używany do wzrostu realistycznego kształtu na poziomie drzewa.

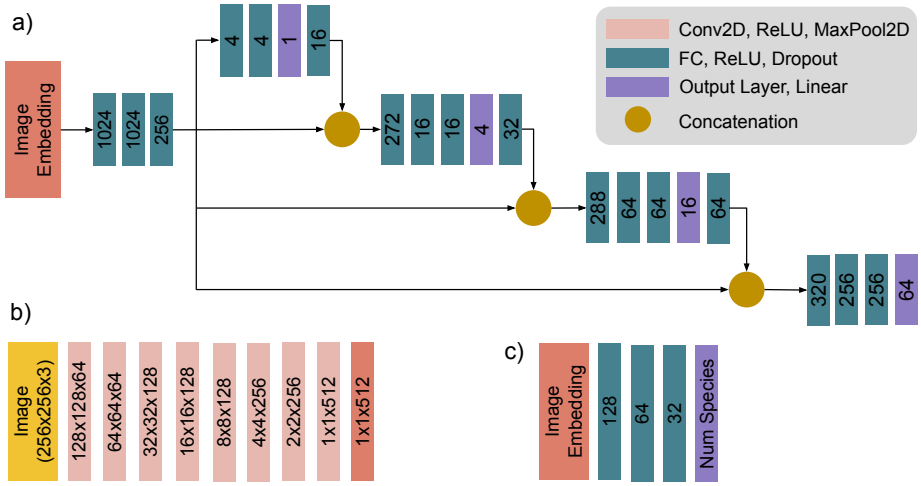

Fig. 2.7 Architektury sieci wykorzystywane w opisanym procesie. Użyto kaskady warstw do przewidywania RBV o czterech różnych rozdzielczościach na bazie wektora osadzenia obrazu (a). Wejściem do tej sieci jest wektor cech obrazu, które są pozyskiwane za pomocą lekkiej architektury ConvNet (b). Do identyfikacji gatunku drzewa stosowana jest architektura ConvNet (b), aby przewidzieć prawdopodobieństwa klasyfikacji przy użyciu kilku dodatkowych warstw (c).

Model drzewa w opisywaniej metodzie jest zdefiniowany jako acykliczny graf $G = \{N, E\}$ składający się z węzłów $n \in N$, krawędzi $e \in E$ oraz zestawu globalnych parametrów P_g. Jeden węzeł w G jest węzłem korzeniowym n_r. Każdy węzeł n przechowuje atrybuty definiujące stan segmentu gałęzi modelu drzewa, takie jak średnica, wiek, pozycja w przestrzeni 3D i siła wzrostu. Rozwój drzewa wyraża się poprzez dodawanie, usuwanie i modyfikowanie wartości atrybutów węzłów podczas symulacji.

Fenomenologiczny wzrost wykorzystuje funkcję wzrostu, która przypisuje nowe stany węzłom na podstawie ich obecnego stanu w każdym kroku symulacji. Funkcja ta opisuje

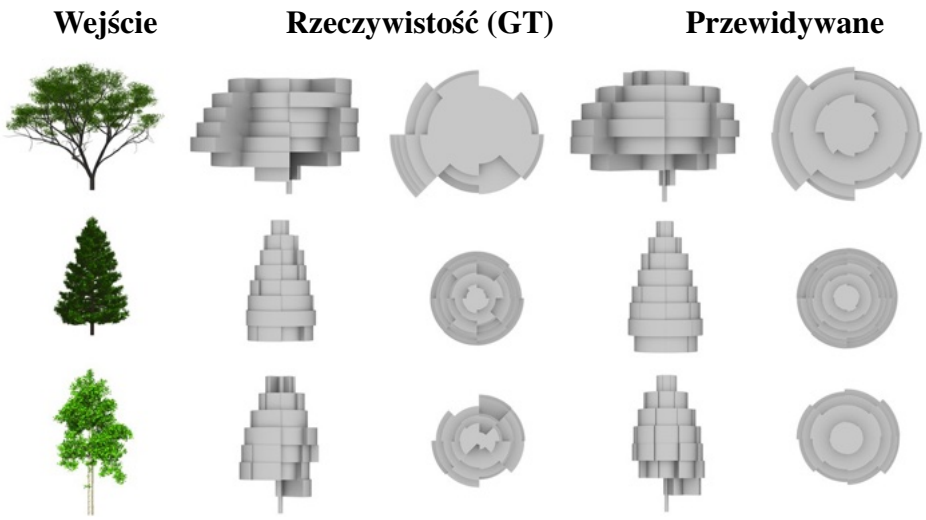

Fig. 2.8 Wizualizacja RBV rzeczywistego (GT) i przewidywanego widoku z góry i z boku. Sieć RBV CNN pozwala dokładnie zrekonstruować strukturę 3D modelu drzewa na podstawie wejściowej maski segmentacji semantycznej.

dobrze zbadane procesy biologiczne, takie jak filotaksja, fototropizm, geotropizm, zrzucanie gałęzi, hormonalna kontrola pąków i powtarzanie wzoru gałęzi. Szczegółowy opis funkcji wzrostu znajduje się w pracach [83, 52].

W przypadku wzrostu samoorganizującego się markery są rozmieszczane A w przestrzeni 3D wokół modelu drzewa. Rosnące gałęzie są przyciągane do markerów w tej przestrzeni, które znajdują się w stożkowej objętości rozciągniętej w kierunku wzrostu. Markery znajdujące się w pobliżu gałęzi są usuwane. To powoduje, że gałęzie unikają wzrostu w te same obszary przestrzeni i wspomaga konkurencję między gałęziami o przestrzeń. Gałęzie teoretycznie mogą się zderzyć, ale w praktyce widać żadnych kolizji. Szczegółowe informacje na temat tego algorytmu znajdują się w pracach [76, 59].

2.4.4 Rekonstrukcja drzew za pomocą wzrostu bi-modalnego

Algorytmy głębokiego uczenia się opisane w sekcji 2.4.2 zwracają maskę semantyczną M drzewa, która obejmuje główne gałęzie w pobliżu pnia oraz listowie. Następnie sieci RBV CNN i sieci Species CNN są użyte do oszacowania wartości RBV oraz identyfikatora gatunku U na podstawie M. Domyślnie oszacowany RBV jest znormalizowany pod względem wysokości pionowej. Aby uwzględnić różnice w wysokości pomiędzy gatunkami, RBV jest skalowane za pomocą oszacowanego identyfikatora gatunku. Identyfikator gatunku U jest używany do wybrania zestawu zdefiniowanych parametrów P dla odpowiadającego gatunku. Następnie modelu rozwoju bi-modalnego generuje model drzewa z parametrami P. W końcu

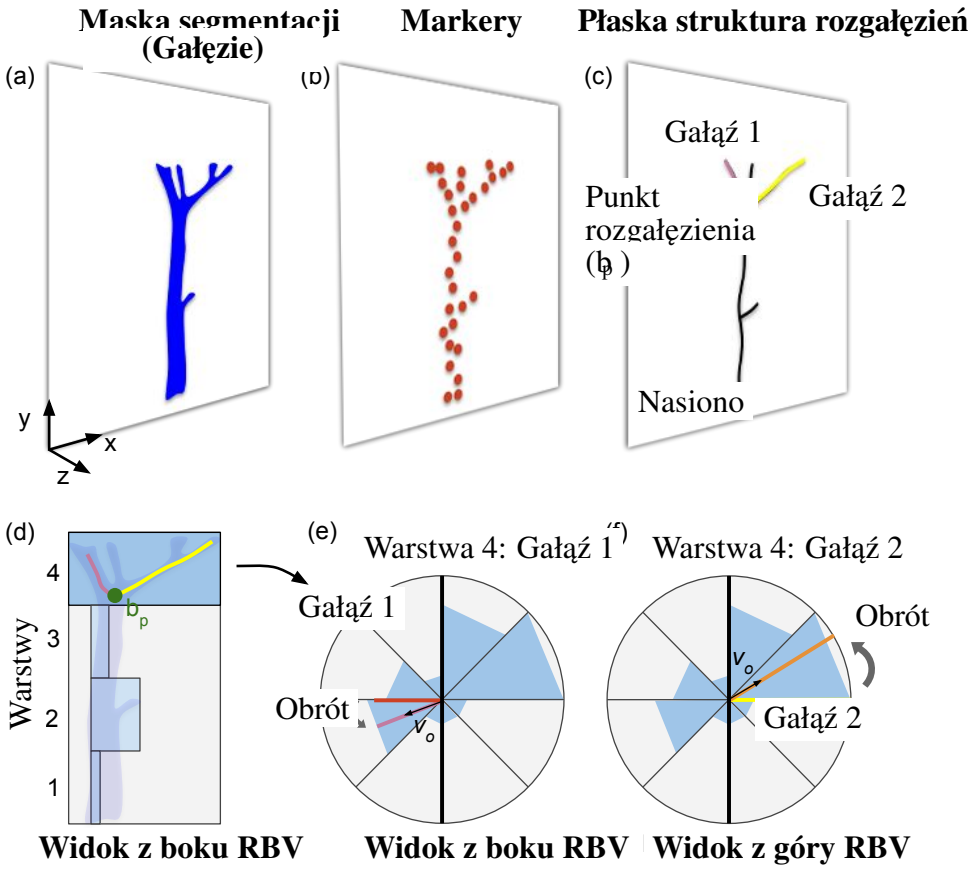

Fig. 2.9 Model wzrostu bi-modalnego ogranicza wzrost drzewa za pomocą maski segmentacji M i RBV. Maska segmentacji gałęzi jest użyta (a) do rozmieszczenia markerów na płaszczyźnie 2D w przestrzeni 3D poprzez próbkowanie maski segmentacji (b). Następnie markery są użyte do kierowania procesem wzrostu struktury rozgałęzień w płaszczyźnie (c). Aby uzyskać strukturę rozgałęzień w przestrzeni 3D, gałęzie są obracane, wyznaczając wektor obrotu na podstawie geometrycznego środka sektorów w RBV (d, e, f). W zależności od tego, gdzie punkt rozgałęzienia jest zlokalizowany w RBV, używana jest albo lewa (e) albo prawa połowa (f) sektorów RBV do obliczenia geometrycznego środka i wynikowego wektora obrotu v_o.

RBV jest używany do przestrzennego ograniczenia wzrostu modelu drzewa, tak aby nie wyrastał on poza RBV (rys. 2.4.3).

Zanim zostanie wygenerowana geometria drzewa za pomocą modelu rozwoju, rozmieszczane są markery A w następujący sposób: po pierwsze, umieszczana jest pionowo zorientowana płaszczyzna I_M, współosiowa z osią x, w przestrzeni 3D i jest tekstowana maską M (rys. 2.9a). Następnie rozmieszczane są markery na płaszczyźnie I_M w miejscach pokrytych odpowiednimi pikselami maski M, które wskazują na zajęcie przez gałęzie (rys. 2.9b). Po tej operacji markery A są rozmieszczone na płaszczyźnie 2D w przestrzeni 3D zgodnie z rozmieszczeniem pikseli w masce M.

Po wypełnieniu sceny 3D markerami, drzewo powstaje za pomocą trybu wzrostu samoorganizującego się modelu rozwoju. Kierunek wzrostu jest inicjowany wektorem w górę i nasiono jest umieszczane w pozycji najniższego markera A. Wzrost trwa, aż wszystkie markery zostaną zużyte, a planar wytworzy strukturę rozgałęzień (rys. 2.9c). Następnie planar jest przestawiany na przestrzeń 3D RBV, aby uzyskać naturalną strukturę rozgałęzień (rys. 2.9d). Dla każdego punktu rozgałęzienia $b_p \in G$ wyznaczana jest warstwa l_i, w której się znajduje. W tej warstwie poziomu i wybierana jest albo lewa, albo prawa połowa sektorów, aby obliczyć geometryczny środek g_c. Następnie obliczany jest wektor obrotu $v_o = b_p - g_c$, względem którego ustawiana jest gałąź wychodzącą z punktu b_p i obracana wokół osi y (rys. 2.9e, f). Ponieważ geometryczny środek wskazuje zakres sektorów, gałęzie są obracane w kierunku największych sektorów dla każdej warstwy. Należy zauważyć, że kolizje są rozwiązane pośrednio, ponieważ gałęzie nie będą rosły w zajętą przestrzeń.

Korona drzewa

Po zakończeniu trybu wzrostu samoorganizującego się i wygenerowaniu początkowej struktury rozgałęzień 3D model przełącza się na tryb wzrostu fenomenologicznego. RBV są używane jako objętości ograniczające — każda wychodząca gałąź jest ścinana. Ten krok jest inspirowany otwartymi systemami L [52], aby umożliwić wzrost gałęzi do ustalonych objętości przestrzennych. Rozwój drzewa jest zakończony, gdy osiągnięta zostanie określona liczba węzłów lub drzewo osiągnie maksymalny wiek. Ostatecznie wygenerowany model drzewa znajduje się całkowicie wewnątrz RBV, a jego projekcja z punktu widzenia odpowiada masce segmentacji M.

2.4.5 Dynamiczne modele drzew

Aby wykazać przydatność opracowanej metody do tworzenia treści, zostaje zaprezentowany sposób, w jaki zrekonstruowane drzewa mogą być łatwo zintegrowane z metodami

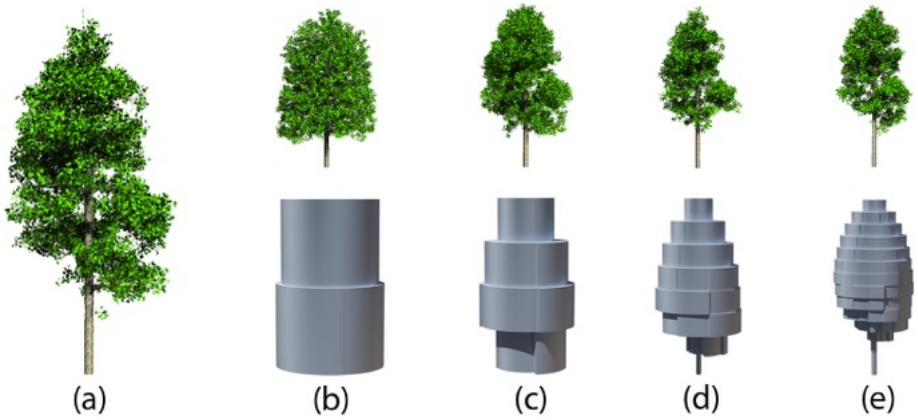

Fig. 2.10 Wyniki modelowania dla różnych poziomów RBV. Użyto drzewa wejściowego (a) i przewidziano RBV z rozdzielczościami 2x2 (b), 4x4 (c), 8x8 (d) oraz 16x16 (e). Wyższa rozdzielczość pozwala na precyzyjniejsze ograniczenie wzrostu, a tym samym bardziej wierne odwzorowanie drzewa wejściowego.

umożliwiającymi modelowanie dynamiki drzew. Do efektywnej symulacji dynamicznego zachowania wyekstrahowanych struktur drzewnych wykorzystano pręty Cosserata [50, 79]. Każda gałąź jest dyskretyzowana poprzez rozmieszczenie kilku węzłów wzdłuż linii środkowej, które są połączone segmentami prętów. Podobnie są łączone sąsiednie gałęzie. Wymaga to przechowywania dodatkowych atrybutów dla każdego węzła n, takich jak prędkość i prędkość kątowa, co pozwala na wygodne implementowanie efektów gięcia i skręcania. Szczegółowe informacje dotyczące wykorzystania dynamiki opartej na pozycjach według Kugelstadta i Schoemera można znaleźć w pracy [61].

2.5 Implementacja i wyniki

Model rozwojowy i RBV zostały zaimplementowane w C++ z użyciem OpenGL, co umożliwia generowannie dużej liczby drzew do tworzenia zbiorów danych. Sieci neuronowe opisane w sekcji 2.4.2 zostały zaimplementowane jako osobne moduły w Pythonie, z użyciem backendu Keras i TensorFlow. W przypadku DeepLab-V3 [12] wykorzystano publicznie dostępne implementacje w Pythonie i TensorFlow.

2.5.1 Trenowanie sieci neuronowej

Do segmentacji semantycznej użyto sieci DeepLab-V3 z architekturą xception41, inicjalizując ją wagami wytrenowanymi na zbiorze danych ImageNet. Wartości stóp atrous zostały ustawione na (6, 12, 18), współczynnik wyjścia na 8, współczynnik uczenia na 0,0001, a wielkość partii na 2 (z uwagi na ograniczenia pamięci). Sieć była trenowana przez

Tabela 2.1 Wskaźniki błędów uzyskane dla różnych konfiguracji sieci i treningu: *Błąd GT* odnosi się do trenowania naszej kaskadowej sieci z maskami typu ground truth. Wyniki trenowania sieci kaskadowej z przewidywanymi maskami z DeepLab-V3 są oznaczone jako *Błąd DL*. *Błąd RM* odnosi się do wyników trenowania naszej sieci na połączonych obrazach RGB i maskach segmentacji semantycznej. W celu walidacji przydatności tej sieci kaskadowej przeprowadzono studium ablacyjne z typową siecią CNN (8 warstw konwolucyjnych i 3 warstwy gęste), trenowaną indywidualnie dla każdej warstwy na syntetycznych maskach typu ground truth. Wskaźniki błędów dla tego eksperymentu są oznaczone jako *Błąd CO*. LxS oznacza liczbę warstw (L) i sektorów (S).

Poziom RBV	LxS	# Param.	Błąd GT	Błąd DL	Błąd RM	Błąd CO
1	1x1	1	1.6%	1.9%	1.7%	1.5%
2	2x2	4	1.7%	1.8%	1.9%	1.9%
3	4x4	16	3.1%	3.3%	3.4%	3.7%
4	8x8	64	3.5%	3.9%	4.0%	4.1%
5	16x16	256	4.2%	4.3%	4.4%	4.6%

60 tysięcy iteracji na 21 tysiącach obrazów wejściowych z syntetycznego zbioru danych, co zajęło około 8 godzin. Sieć osiąga średnią wartość IoU na poziomie $74,7\%$ dla wszystkich klas i wymaga około 3,2 sekundy na interferencję pojedynczej, wysokiej rozdzielczości ($1,280 \times 1,280$ pikseli) maski segmentacji.

Sieć RBV CNN była trenowana za pomocą optymalizatora Adam, współczynnika uczenia $0,001$ oraz partii wielkości 32, z wykorzystaniem straty Huber dla regresji [31] na masce segmentacji M oraz RBV R z opisanego w sekcji 2.4.2 zbioru danych. Maski segmentacji (podzielone na trzy kanały RGB), oraz wartości RBV są znormalizowane do zakresu $[0,1]$.

Tab. 2.1 przedstawia średni błąd absolutny dla różnych eksperymentów: *Błąd GT* odnosi się do trenowania kaskadowej sieci z użyciem masek segmentacji semantycznej typu ground truth (danych wzorcowych). Trenowanie sieci kaskadowej z przewidywanymi maskami z DeepLab-V3 jest przedstawione jako *Błąd DL*. *Błąd RM* oznacza wyniki trenowania sieci na połączonych obrazach RGB i maskach segmentacji semantycznej. W celu dalszej walidacji przydatności sieci kaskadowej przeprowadzono studium ablacyjne z typową siecią CNN (8 warstw konwolucyjnych i 3 warstwy gęste), która była trenowana indywidualnie dla każdej warstwy na syntetycznych maskach typu ground truth. Wskaźniki błędów dla tego eksperymentu przedstawiono jako *Błąd CO*. Podczas gdy względny błąd pomiędzy *Błąd GT* i *Błąd DL* (0,4% na poziomie 4) ujawnia błąd wprowadzony przez sieć segmentacji semantycznej, względny błąd pomiędzy *Błąd GT* i *Błąd CO* (0,3% na poziomie 4) pokazuje, że architektura sieci kaskadowej jest lepsza od typowego schematu trenowania.

Sieć była trenowana przez 4 godziny, co pozwoliło na uzyskanie wartości RBV w czasie 17 ms. Sieć klasyfikująca gatunki była również trenowana z użyciem optymalizatora Adam, ze współczynnikiem uczenia 0,001 oraz partii wielkości 32. Sieć była trenowana z użyciem funkcji straty entropii krzyżowej na zbiorze danych z siedmioma gatunkami jako

osobnymi klasami, osiągając dokładność identyfikacji gatunku na poziomie 94%. Ta sieć była trenowana przez 1 godzinę, uzyskując identyfikatory gatunków w czasie 12 ms. Identyfikator gatunku jest następnie używany do wyszukania zdefiniowanego zestawu parametrów P dla każdego gatunku. Każda z trzech sieci jest trenowana oddzielnie, ale na tym samym zbiorze danych. Całkowity czas potrzebny do automatycznej rekonstrukcji modelu drzewa 3D w opisanej metodzie wynosi średnio 10 sekund.

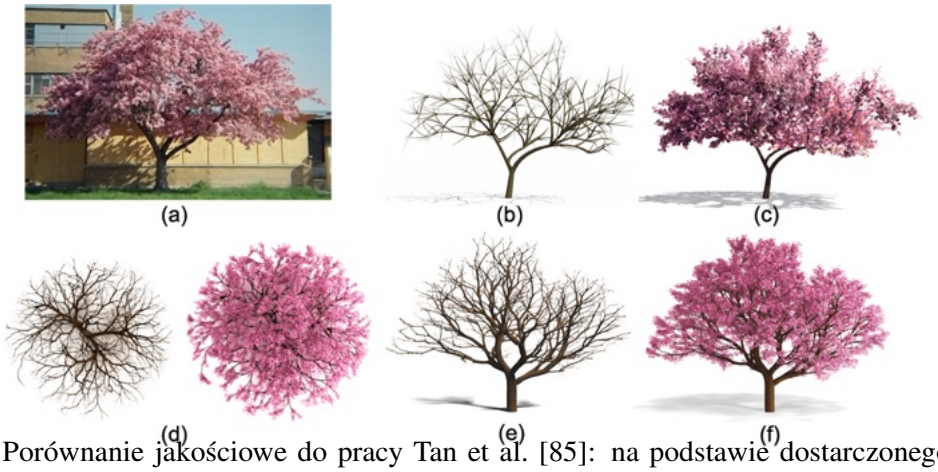

Fig. 2.11 Porównanie jakościowe do pracy Tan et al. [85]: na podstawie dostarczonego zdjęcia drzewa wiśni (a) oraz zrekonstruowanego modelu (b, c), rekonstrukcja podobnego modelu drzewa za pomocą przedstawionej metody (d-f), o zbliżonych cechach wizualnych. W odróżnieniu od poprzedniej metody, która wymagała ręcznego szkicowania maski segmentacji przez użytkownika, opisana metoda automatycznie rekonstruuje model drzewa w mniej niż 3 sekundy.

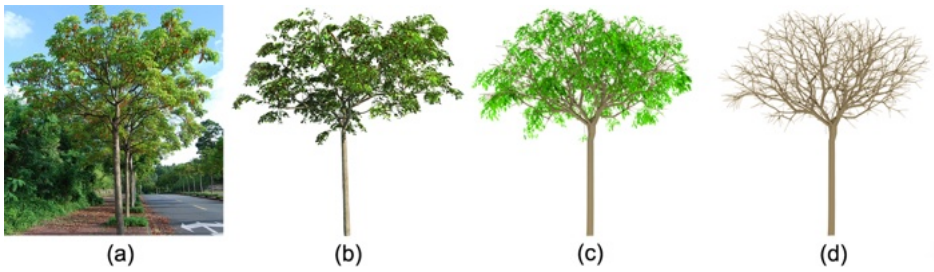

Fig. 2.12 Porównanie jakościowe do pracy Livny et al. [46]: podczas gdy ta metoda rekonstruuje model drzewa (b) z chmur punktów 3D, tu użyto tylko pojedynczego zdjęcia (a) do rekonstrukcji modelu drzewa (c, d).

2.5.2 Wyniki

Rysunki 2.13 oraz 2.14 przedstawiają rekonstrukcję rzeczywistych i syntetycznych drzew za pomocą przedstawionej metody. W obu przypadkach użyto sieci segmentacji semantycznej

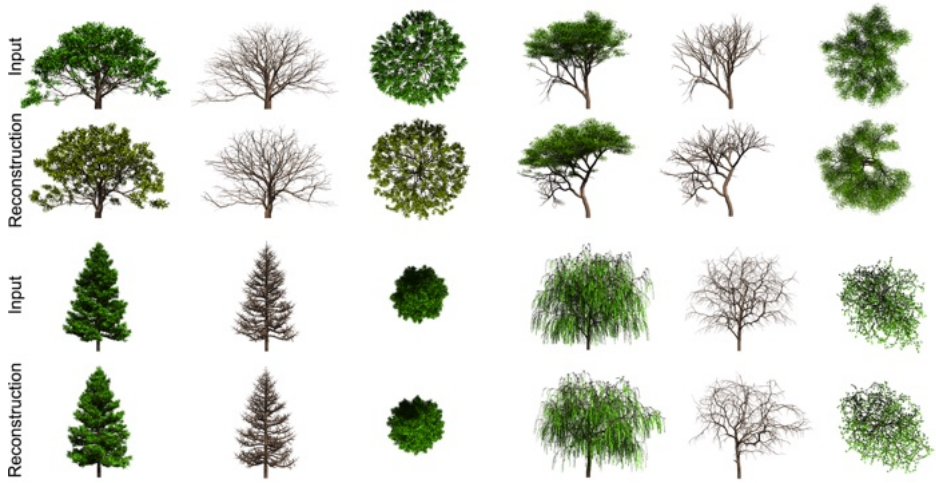

Fig. 2.13 Rekonstrukcje syntetycznie generowanych modeli drzew czterech różnych gatunków: dąb (lewy górny), akacja (prawy górny), sosna (lewy dolny), i wierzba (prawy dolny). Metoda opisana w tej pracy umożliwia rekonstrukcję głównej struktury rozgałęzień oraz ogólnego kształtu drzewa z wysoką wiernością wizualną. Dla każdego modelu wejściowego drzewa przedstawiane są dwa zrekonstruowane modele wygenerowane z różnymi losowymi nasionami oraz bi-modalnym wzrostem z RBV i maską segmentacji.

do wykrywania i segmentacji drzewa na pojedynczym zdjęciu. Następnie wykorzystano maskę do identyfikacji gatunku drzewa, co z kolei dostarcza parametrów modelu proceduralnego. Dodatkowo maska jest używana do oszacowania RBV drzewa. Następnie wykorzystano maskę, RBV oraz parametry modelu proceduralnego do generowania struktur rozgałęzień podobnych do drzewa przedstawionego na obrazie wejściowym. W zależności od użytego gatunku, liczba gałęzi zrekonstruowanych modeli waha się od 2 tys. do 10 tys.

Rysunek 2.10 pokazuje, jak różne poziomy RBV wpływają na wzrost drzewa. Na podstawie syntetycznego drzewa o znanej geometrii (rys. 2.10a) budowanych jest kilka RBV o różnych rozdzielczościach (2x2, 4x4, i 8x8). Następnie użyto modelu rozwojowego do wzrostu drzewa przy użyciu tych RBV.

Przedstawiona metoda pozwala na generowanie złożonych i szczegółowych struktur rozgałęzień, które można bezpośrednio wykorzystać do animacji. Dla wyniku pokazanego na rysunku 2.16 symulowana jest dynamika prętów [61], aby animować model drzewa na podstawie interakcji użytkownika. Użytkownik pociąga gałąź, a następnie ją puszcza, co powoduje typowe ruchy kołysania drzew. Na podstawie metody opisanej w tym rozdziale możliwe jest zrekonstruowanie drzew i natychmiastowe użycie ich jako gotowych do animacji treści w typowych procesach produkcji.

Rysunek 2.17 pokazuje, że RBV są zdolne do reprezentowania złożonych asymetrycznych struktur rozgałęzień. Dla obu modeli drzew, dębu i sosny, odpowiednie RBV uchwytują

Fig. 2.14 Przykłady różnorodnych rekonstrukcji dla czterech różnych rzeczywistych drzew. Użyto sieci neuronowej do segmentacji semantycznej, aby uzyskać maski dla gałęzi i liści ze zdjęć (a)-(c). Następnie wykorzystano nową reprezentację radialnej objętości ograniczającej (RBV) wraz z algorytmem bi-modalnego wzrostu do rekonstrukcji realistycznych struktur rozgałęzień (d). Zrekonstruowane drzewa wykazują cechy wizualne podobne do tych, które można zaobserwować na zdjęciach (e, f).

główne cechy struktury rozgałęzień, co jest istotną właściwością przy rekonstrukcji modelu drzewa z pojedynczego obrazu.

Rysunki 2.11 i 2.12 przedstawiają jakościowe porównanie przedstawionych wyników do najnowszych metod rekonstrukcji drzew. W przypadku rekonstrukcji drzew z pojedynczego obrazu pokazano jeden z wyników z pracy Tan et al. [85]. Na podstawie pojedynczego zdjęcia drzewa wiśni, ich metoda rekonstruuje szczegółowy model drzewa, dostarczając szkice struktury głównych rozgałęzień i kształtu korony. W odróżnieniu od nich, użyto fotografii do automatycznej rekonstrukcji modelu drzewa w zaledwie kilka sekund. Rysunek 2.12 pokazuje porównanie do metody Livny et al. [46], która polega na zestawach punktów zeskanowanych laserowo - co jest mniej wygodne do uzyskania niż pojedyncze obrazy - do rekonstrukcji szczegółowych modeli drzew.

Na rysunku 2.18 pokazane są dwa przypadki niepowodzenia. W pokazanych przykładach fotografii (a) sieć segmentacji semantycznej nie była w stanie w pełni oddzielić pikseli drzewa na pierwszym planie od tła i innych pikseli drzew (b). W konsekwencji, sieć RBV nie była w stanie prawidłowo przewidzieć odpowiedniego RBV (c), co z kolei doprowadziło do rekonstrukcji, które nie odpowiadają kształtowi drzewa na zdjęciu (d).

W końcu, na rysunku 2.15 pokazany jest eksperyment dotyczący rekonstrukcji wielu drzew. Ponieważ niniejsza metoda opiera się na identyfikacji masek segmentacji semantycznej drzew w celu oddzielenia pikseli gałęzi od pikseli liści, najpierw wykrywane

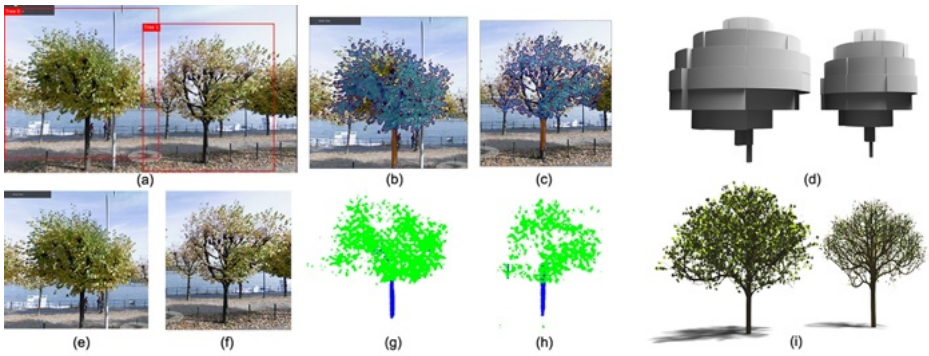

Fig. 2.15 Rekonstrukcje wielu drzew: aby zrekonstruować wiele drzew, najpierw wykrywane są ramki ograniczające drzewa (a), w celu uzyskania przyciętych obrazów (e, f). Dla każdego przyciętego obrazu można następnie obliczyć maski segmentacji semantycznej (g, h) — nakładane na obraz RGB w (b, c). Następnie przewidywane są RBV (d), aby zrekonstruować modele drzew (i). Należy zauważyć, że proces ten nie dąży do automatycznej rekonstrukcji scen; układ 3D jest definiowany ręcznie.

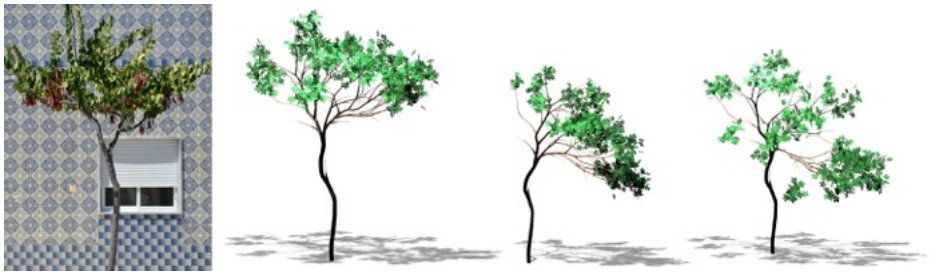

Fig. 2.16 Rekonstrukcja modelu drzewa z fotografii oraz symulacja realistycznych ruchów kołysania gałęzi za pomocą "Cosserat rods" [61]. Użytkownik pociąga gałąź, a następnie ją puszcza, co skutkuje charakterystycznymi ruchami kołysania struktur rozgałęzień.

są ramki ograniczające drzewa (a) za pomocą sieci do detekcji obiektów [73]. Następnie indywidualnie stosana jest niniejsz metoda do każdego z wygenerowanych przyciętych obrazów (e, f), aby najpierw uzyskać maski segmentacji semantycznej (g, h), które są nakładane na obraz RGB dla odniesienia (b, c). Maski segmentacji mogą następnie być użyte do wygenerowania RBV (d) i zrekonstruowanych modeli drzew (i). Należy zauważyć, że celem nie dążenie do automatycznej rekonstrukcji całych scen, dlatego drzewa muszą być pozycjonowane ręcznie.

2.6 Ewaluacja, dyskusja i ograniczenia

Metoda jest oceniana poprzez zestaw pomiarów zaprojektowanych do oceny podobieństwa formy drzew. W tym celu użyto kilku pomiarów rozwoju stosowanych w leśnictwie do kwantyfikacji drzew [8, 91]. Obejmują one średni kąt rozgałęzienia, średnicę pnia (u

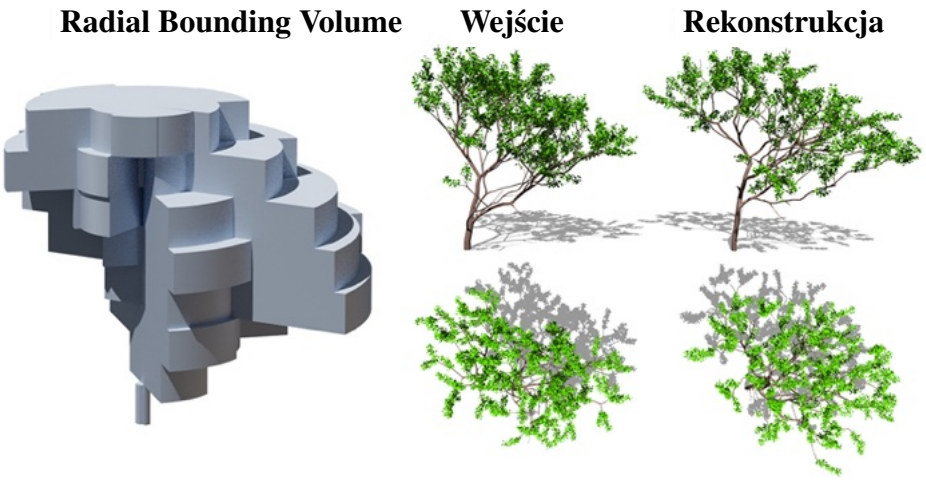

Fig. 2.17 RBV mogą być używane do reprezentowania i rekonstrukcji skomplikowanych struktur rozgałęzień, wymaganych w asymetrycznych modelach drzew.

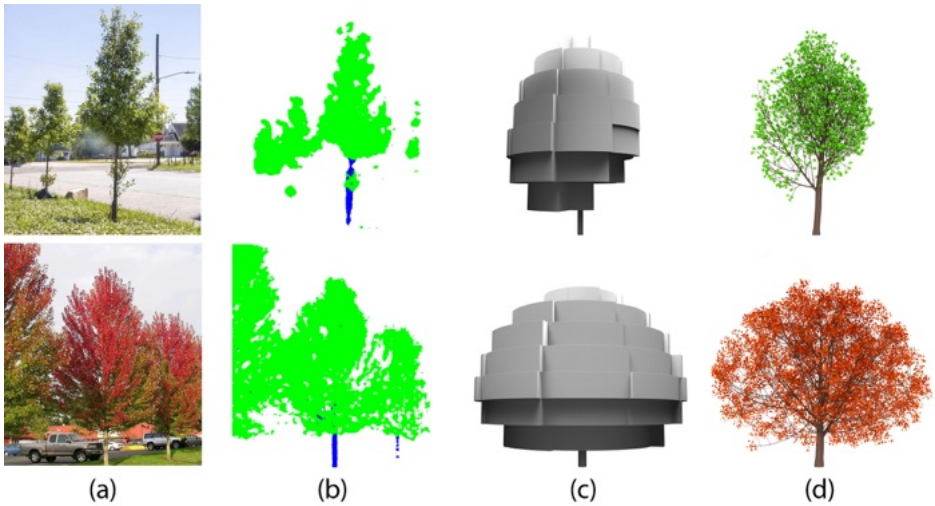

Fig. 2.18 Dwa przypadki niepowodzenia: na pokazanych zdjęciach (a), sieć segmentacji semantycznej nie była w stanie w pełni oddzielić pikseli drzewa na pierwszym planie od tła i innych pikseli drzewa (b). W konsekwencji, sieć RBV nie była w stanie prawidłowo przewidzieć odpowiedniego RBV (c), co z kolei doprowadziło do rekonstrukcji, które nie odpowiadają kształtowi drzewa na zdjęciu (d).

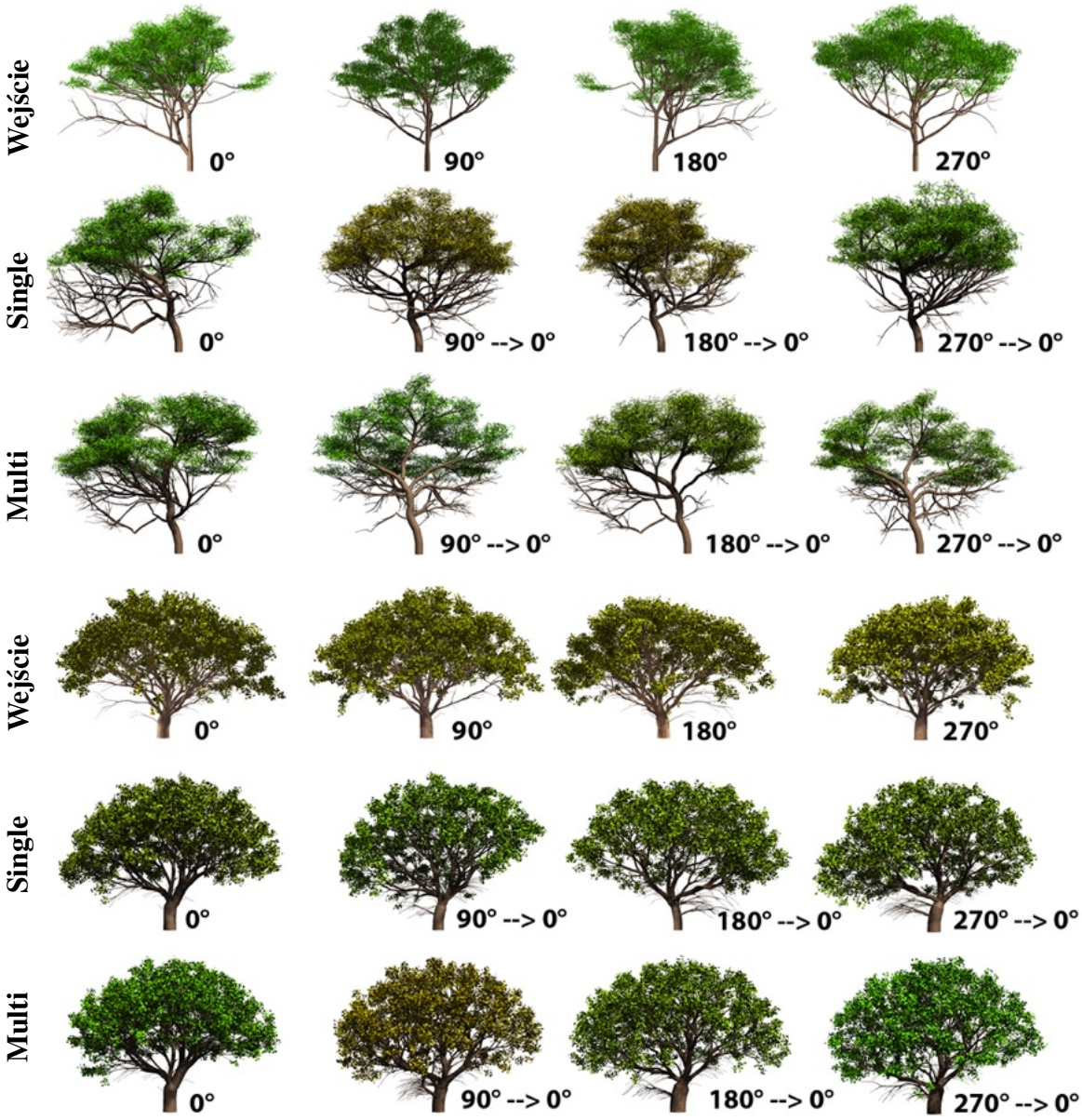

Fig. 2.19 Wynik eksperymentu na niezmienność widoku. Pokazano drzewa z czterech punktów widzenia przy 0°, 90°, 180°i 270°rotacji. Następnie zrekonstruowano modele i pokazano je z kąta 0°. Celem jest wykorzystanie tej metody do spójnej rekonstrukcji modeli z każdego widoku, aby przypominały model wejściowy oglądany z kąta 0°. Wiersze *input* pokazują modele drzew wejściowych z czterech różnych punktów widzenia. Drzewa w wierszach *single* pokazują rekonstrukcje uzyskane z RBV CNN, która widziała tylko dane wejściowe z jednego widoku, podczas gdy drzewa pokazane w wierszach *multi* były rekonstruowane na podstawie RBV CNN trenowanej na wielu widokach dla każdego drzewa.

podstawy), wysokość drzewa, całkowitą biomasę oraz wskaźnik powierzchni liści (LAI). Całkowita biomasa jest uzyskiwana poprzez obliczanie całkowitej objętość wszystkich gałęzi. LAI jest definiowane jako projekcja geometrii drzewa na powierzchnię ziemi na jednostkę powierzchni.

Użyto tych pomiarów do kwantyfikacji podobieństwa między modelami drzew. Użyto modeli drzew typu ground truth (GT) i porównano je z zrekonstruowanymi modelami drzew z masek segmentacji typu ground truth i RBV (Rec A). Dodatkowo obliczane są modele drzew na podstawie przewidywanych masek segmentacji i RBV, które są uzyskiwane z wspomnianych sieci neuronowych (Rec B). Na rysunku 2.20 pokazane są wykresy skrzynkowe wszystkich używanych pomiarów rozwoju dla wszystkich gatunków w syntetycznym zbiorze danych. Większość wykresów skrzynkowych wskazuje na podobne średnie i odchylenia standardowe między danymi GT i Rec A oraz GT i Rec B. Ogólna zgodność wyników GT i Rec A wskazuje, że bi-modalny model rozwojowy produkuje architektury drzew podobne do tych generowanych wyłącznie w trybie fenomenologicznym. Jednak nie wszystkie rekonstrukcje wykazują wysoki stopień pokrywania się. Najbardziej zauważalne są rozbieżności w rozkładach biomasy dla klonu i dębu, wynikające z różnych dynamik wzrostu wymuszonych przez tryb samoorganizacji. Blisko pokrywające się rozkłady wyników GT i Rec B pokazują ogólną wydajność sieci neuronowych na danych syntetycznych.

Ponieważ pomiary kąta rozgałęzień, wysokości drzewa, średnicy pnia, wskaźnika powierzchni liści oraz biomasy reprezentują globalne miary kształtu drzewa, dodatkowo użyto przestrzennej miary lokalnej do oceny geometrii drzew. Nazywano ją maksymalną radialną odległością od pnia (MRT). MRT jest obliczana dla kilku dyskretnych warstw pionowych przez znalezienie maksymalnie odległego węzła gałęzi od pnia w każdej warstwie. Rysunek 2.21 pokazuje średnie wartości MRT dla wszystkich gatunków oraz ich błąd standardowy. Te wykresy dają wizualne wrażenie ogólnej wariancji i średniej geometrii drzew używanych do przygotowania syntetycznego zbioru danych treningowych.

Ponadto, użyto globalnych pomiarów (kąt rozgałęzień, wysokość drzewa, średnica pnia, wskaźnik powierzchni liści, biomasa) razem z pięcioma pionowymi warstwami MRT do skonstruowania 10-wymiarowej przestrzeni oceny podobieństwa. Pięć osi w tej przestrzeni odpowiada pięciu globalnym pomiarom, podczas gdy pozostałe pięć osi jest używanych do reprezentowania MRT dla pięciu jednolicie wybranych warstw pionowych. Następnie zbiór danych treningowych i oszacowany zbiór danych są osadzane w tej przestrzeni wektorów. Odległości w tej przestrzeni wskazują na podobieństwo formy drzewa między dwoma modelami drzew. Na rysunku 2.22 pokazane są wyniki nieliniowej redukcji wymiarów za pomocą t-SNE, aby uzyskać 2-wymiarowe projekcje 10-cio wektorowej przestrzeni.

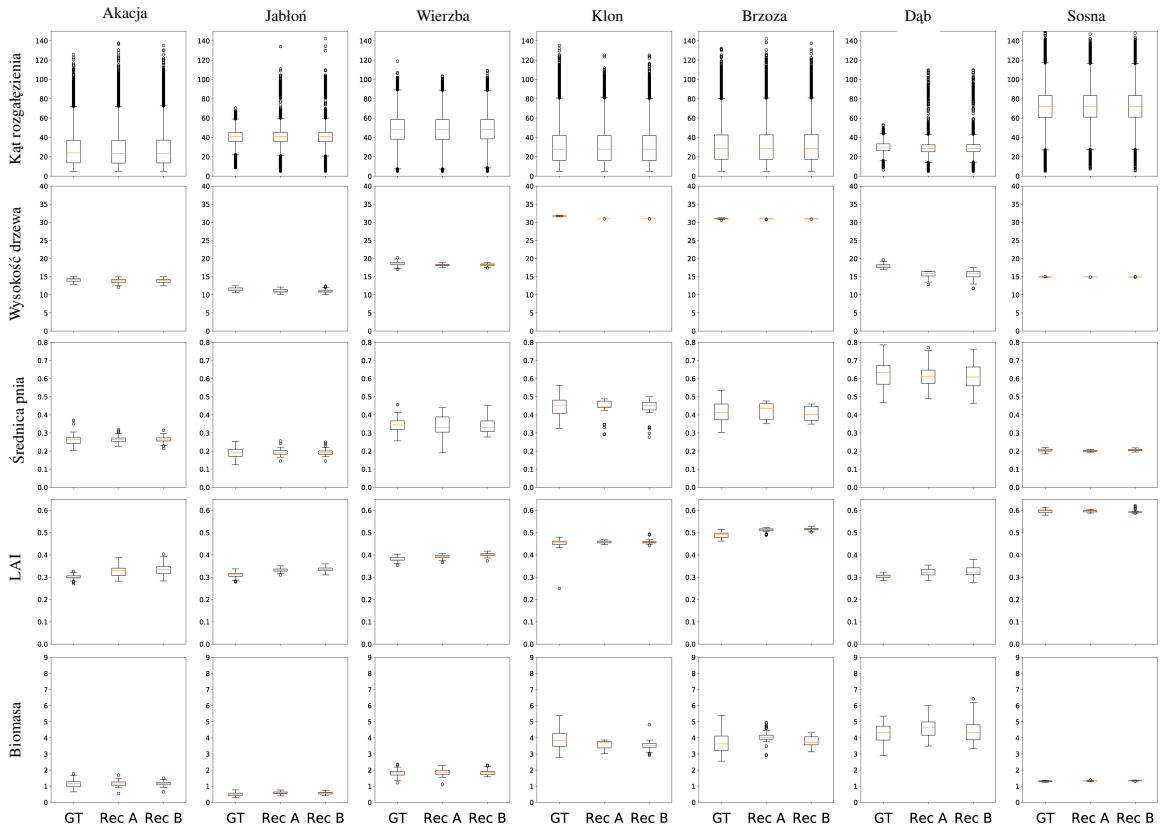

Fig. 2.20 Wykresy skrzynkowe pomiarów rozwoju drzew: oceniane jest podobieństwo formy 3D drzew między danymi ground truth (GT), rekonstrukcjami z masek segmentacji GT i RBV (Rec A) oraz rekonstrukcjami z przewidywanych masek segmentacji i RBV (Rec B). Wiersze oznaczają pomiary formy drzew, takie jak kąt rozgałęzienia, wysokość drzewa, średnica pnia, LAI i biomasa. Kolumny oznaczają wirtualne gatunki używane do tworzenia syntetycznego zbioru danych, w tym akację, jabłoń, wierzbę, klon, brzozę, dąb i sosnę. Ogólnie rzecz biorąc, zgodności średnich i wariancji rozkładów są bardzo zbliżone między danymi GT a danymi rekonstruowanymi.

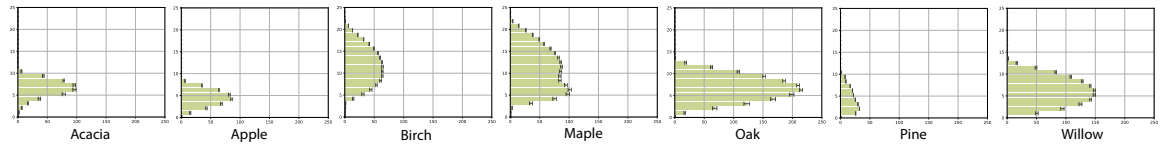

Fig. 2.21 MRT dla wszystkich gatunków: oceniane są wariancje i ogólny kształt modeli drzew w syntetycznym zbiorze danych użytym do trenowania sieci neuronowych. Oś pionowa wskazuje wysokość drzewa, oś pozioma - średnią maksymalną odległość węzłów gałęzi od pnia. Pionowe paski oznaczają odchylenie standardowe dla danego gatunku.

2.6.1 Badanie użytkowników

Przeprowadzono badanie użytkowników, aby zweryfikować, czy zrekonstruowane drzewa są postrzegane jako podobne. Pokazano źródłowe drzewo i zrekonstruowane drzewa 100 użytkownikom. Zadano uczestnkom pytanie: "Czy drzewa wyglądają podobnie?" i musieli wybrać 0-zdecydowanie nie, 1-nie, 2-raczej nie, 3-raczej tak, 4-tak, 5-zdecydowanie tak. Każdej odpowiedzi przypisano wartość $0 - 0.2 - 0.4 - 0.6 - 0.8 - 1$. Przeprowadzono dwa testy: jeden z liśćmi i jeden z tymi samymi modelami bez liści. Przeprowadzono również oddzielne próby dla rekonstrukcji z maską i bez maski. Testy były przeprowadzane dla każdego gatunku, używając dwóch obrazów wybranych z centrum klastra i dwóch obrazów, które były daleko od centroidu (patrz rysunek 2.22). Hipoteza zakładała, że drzewa znajdujące się bliżej centrum będą oceniane jako bardziej podobne niż te oddalone od centrum. Do przeprowadzenia badania użyto platformy Mechanical Turk, wybierając wyłącznie certyfikowanych użytkowników Mechanical Turk Masters, aby zapewnić wiarygodność uzyskanych odpowiedzi.

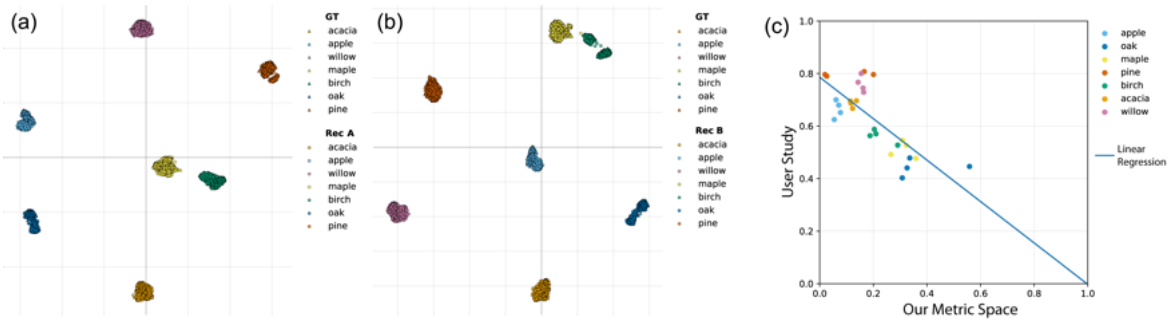

Fig. 2.22 Wykresy t-SNE dla: 10-wymiarowej przestrzeni skonstruowanej z LAI, wysokości drzewa, biomasy, średnicy pnia, kąta rozgałęzień i pięciu pionowych warstw MRT. Modele drzew typu ground truth (GT) są przedstawione jako kolorowe dyski, modele drzew Rec A (a) i Rec B (b) jako kolorowe trójkąty. Ogólnie rzecz biorąc, rozkłady GT, Rec A i Rec B modeli drzew pokrywają się dla każdego gatunku, co wskazuje, że niniejszy algorytm rekonstrukcji umożliwia wierne odtworzenie modeli drzew. W (c) pokazany jest wykres z wyników badania użytkowników, pokazujący ranking podobieństwa GT do Rec B (od 0 do 5) oraz odległości w znormalizowanej przestrzeni pomiarów. Jak pokazuje linia regresji liniowej, podobieństwa są negatywnie skorelowane. Oznacza to, że modele drzew, które są dalej od siebie w 10-wymiarowej przestrzeni, otrzymały średnio niższy wynik w badaniu użytkowników. Dlatego korelacja badania użytkowników i nakładanie się rozkładów w 10-wektorowej przestrzeni pokazuje, że przedstawiona metoda generuje wizualnie podobne rekonstrukcje.

Z liśćmi: Użytkownicy wybrali drzewa jako podobne do źródła w 63% przypadków, z lekką preferencją dla drzew daleko od centrum 64% w porównaniu do drzew blisko centrum 61%. Jeśli maska była używana do rekonstrukcji, postrzegane podobieństwo wynosiło 63%. Rekonstrukcja bez maski była na poziomie 61%.

Bez liści: Użytkownicy wybrali drzewa jako podobne do źródła w 59% przypadków, bez preferencji co do odległości drzewa od centrum. Nie było preferencji dla rekonstrukcji z i bez maski (58.0 vs. 57.8).

Na rysunku 2.22 (c) pokazany jest wykres z wyników badania użytkowników, pokazujący ranking podobieństwa GT do Rec B (od 0 do 5) oraz odległości w znormalizowanej 10-cio wektorowej przestrzeni. Jak pokazuje linia regresji liniowej, podobieństwa są negatywnie skorelowane. Oznacza to, że modele drzew, które są dalej od siebie w tej przestrzeni, otrzymały średnio niższy wynik w badaniu użytkowników i odwrotnie. Dlatego korelacja badania użytkowników i nakładanie się rozkładów w przestrzeni pomiarów pokazuje, że przedstawiona metoda generuje wizualnie podobne rekonstrukcje.

2.6.2 Dyskusja i ograniczenia

Istnieją trzy odrębne metody generowania modeli 3D drzew rzeczywistych. Po pierwsze, drzewa mogą być rekonstruowane na podstawie danych czujników, takich jak skanery laserowe [46] lub wiele obrazów [54, 86]. Te metody są zazwyczaj bardziej dokładne niż przedstawione podejście, ale opierają się na danych, które są kosztowne do uzyskania i nie są łatwo dostępne w porównaniu z pojedynczymi zdjęciami drzew. Po drugie, zaproponowano różne proceduralne lub interaktywne podejścia do wzrostu drzew, aby generować wiarygodne architektury 3D drzew (e.g. xfrog, SpeedTree). Jednak ręczne dostrajanie parametrów nie jest intuicyjne i wymaga specjalistycznej wiedzy. Ponadto, ręczne dostrajanie parametrów do rekonstrukcji drzew przedstawionych na zdjęciach wejściowych brakuje kontroli i nie jest wykonalne w aplikacjach, które wymagają generowania dużej liczby różnorodnych modeli drzew 3D. Wreszcie, istnieją półautomatyczne metody rekonstrukcji drzew z pojedynczego obrazu, takie jak proponowane przez Tan et al. [85], które opierają się na adnotacjach użytkownika. Te metody są najbliżej związane z przedstawionym podejściem, ale ponieważ polegają na ręcznie generowanych maskach, nie wspierają rekonstrukcji modeli drzew 3D na dużą skalę.

Niniejsza metoda jest pierwszą, która eksploruje w pełni automatyczną rekonstrukcję drzew z pojedynczych zdjęć. Jakość generowanych modeli drzew za pomocą przedstawionej metody jest wysoka wizualnie. Metoda ta ma jednak kilka ograniczeń. Przetestowiono szeroki zakres strategii augmentacji danych, które obejmują konfigurowanie parametrów renderowania, takich jak dostosowanie pozycji kamery, intensywność cieniowania lub liczba świateł, do transformacji obrazu, takich jak zmiany jasności czy koloru. Augmentacja danych pomaga uzyskać maski dla dużej liczby rzeczywistych gatunków drzew.

Pokonanie luki domenowej poprzez trenowanie sieci neuronowych na danych syntetycznych, aby działały na rzeczywistych fotografiach dowolnych gatunków, pozostaje

wyzwaniem. Ponadto, użyto masek segmentacji semantycznej - zamiast obrazów RGB - do trenowania sieci RBV i sieci klasyfikującej gatunki, aby uniknąć nadmiernego dopasowania sieci do danych syntetycznych. Maski semantyczne kodują istotne cechy modeli drzew, co jest wystarczające do uzyskania identyfikatora gatunku i wartości RBV. Maski semantyczne służą więc jako reprezentacja, która pomaga pokonać lukę domenową między danymi rzeczywistymi a syntetycznymi.

Innym ograniczeniem niniejszej metody jest to, że nie można sensownie zakodować drzew z nieciągłymi poziomymi strukturami rozgałęzień. RBV przechowują tylko jedną wartość odległości na sektor, co ogranicza kodowanie bardziej złożonych kształtów drzew. Tę reprezentację można by rozszerzyć, przechowując wiele wartości (np. dla zakresów) lub dyskretyzując przestrzenny zasięg na kilka części. Jednak rozszerzenie reprezentacji spowodowałoby zwiększenie liczby wartości, które trzeba wyuczyć, co - potencjalnie - mogłoby obniżyć dokładność rekonstrukcji.

2.7 Wnioski

Przedstawiona została nowa metoda rekonstrukcji drzew z pojedynczych fotografii. Umożliwia to rekonstrukcję modeli drzew na dużą skalę, co jest kluczowe dla wielu zastosowań w rekonstrukcji urbanistycznej lub grach i filmach. W odróżnieniu od wcześniejszych prac, przedstawione podejście jest w pełni automatyczne i nie wymaga żadnej interwencji użytkownika. To jest krok naprzód w modelowaniu drzew na skalę masową w grafice komputerowej. Zamiast używania szkiców definiowanych przez użytkownika, wykorzystano sieci neuronowe do uzyskania masek semantycznych drzew, identyfikacji ich gatunku oraz oszacowania ich struktury 3D na podstawie zdjęć 2D. Przedstawiono *radialne objętości ograniczające* jako lekką i stałej wielkości reprezentację modeli drzew, która ułatwia naukę kształtu drzew. Wzrost drzew do RBV przy jednoczesnym ograniczaniu ich rozwoju za pomocą uzyskanych masek semantycznych umożliwia staranne kierowanie wzrostem drzew w celu uchwycenia definiujących szczegółów rzeczywistych drzew. Przeprowadzono obszerną ewaluację zaprezentowanej metody na podstawie szeregu ilościowych metryk oceniających formę drzew. Eksperymenty wskazują, że można z powodzeniem zrekonstruować złożone struktury rozgałęzień 3D w różnych przypadkach drzew i różnych gatunkach drzew. Wreszcie, pokazano, że wygenerowane struktury rozgałęzień mogą być bezpośrednio animowane za pomocą istniejących metod dynamiki prętów. W ten sposób powstała kompleksowa metodę dla twórców treści, która pozwala na tworzenie gotowych do animacji roślin z fotografii.

Biorąc pod uwagę obecny stan tej metody, istnieje wiele możliwości rozszerzenia tematu. Zamiast skupiać się tylko na całych drzewach, obiecujące wydaje się dokładniejsza analiza poszczególnych organów w celu określenia ich cech.

Rozdział 3

Przewidywanie rozmiaru liścia

3.1 Wstęp

W ostatnich latach integracja algorytmów wizji komputerowej w rolnictwie odegrała kluczową rolę w zwiększaniu wydajności i zrównoważonego rozwoju. Jednym z głównych ograniczeń algorytmów uczenia maszynowego jest ich zależność od precyzyjnych i rozbudowanych danych treningowych. Zbieranie odpowiednio oznakowanych danych z rzeczywistego świata, szczególnie do zadań takich jak analiza liści, jest zazwyczaj kosztowne i czasochłonne [90].

Zaprezentowano LAESI, syntetyczny zbiór danych liści, oraz dwa modele proceduralne do jego generowania. Pierwszy to model proceduralny papieru milimetrowego, a drugi do generowania kształtów liści. Przedstawiona metoda wykorzystuje ControlNet [97] do poprawy realizmu wizualnego renderów, podobnie jak w pracy Anagnostopoulou et al. [1]. Zastosowanie wydajnych obliczeniowo modeli proceduralnych w połączeniu z modelami SI generatywnej umożliwia w pełni automatyczne, kontrolowane i masowe generowanie syntetycznych danych, które są użyteczne do trenowania modeli głębokiego uczenia do zadań wizji komputerowej.

Zbiór danych LAESI dostarcza adnotacje w postaci masek semantycznych i etykiet powierzchni liści. Pokazano użyteczność tego zbioru danych, trenując modele wizji komputerowej do przewidywania powierzchni liści i segmentacji semantycznej. Ponadto, porównano modele wizji komputerowej trenowane z różnymi mieszankami danych syntetycznych i rzeczywistych z modelem bazowym trenowanym na 1,7 tys. rzeczywistych obrazów z adnotacjami.

Niniejszy proces generowania syntetycznych obrazów liści LAESI obejmuje kilka etapów (Fig.3.1): *(1) Proceduralne Generowanie Tła z Papieru Milimetrowego:* generowanie tekstur papieru i układów siatek w celu zapewnienia spójnej skali odniesienia; *(2) Proceduralny*

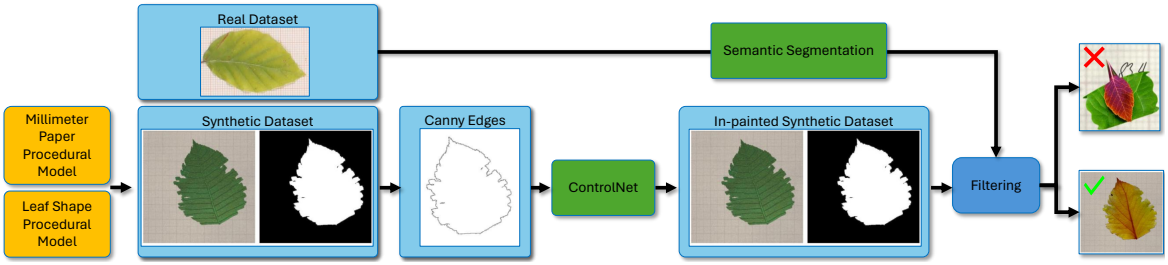

Fig. 3.1 Model procesu LAESI: *Proceduralne Generowanie Tła z Papieru Milimetrowego i Kształtu Liścia* generuje różnorodne tekstury papieru, układy siatek oraz zakresy kształtów, rozmiarów i tekstur liści. *Renderowanie i Ostateczna Kompozycja Syntetycznego Zbioru Danych* łączy liście z tłem, dodając realistyczne oświetlenie i generując adnotacje, takie jak maski semantyczne, etykiety powierzchni oraz krawędzie Canny'ego. *Wypełnianie Zbioru Danych* z wykorzystaniem procesu ControlNet do wypełniania krawędzi Canny'ego generuje obrazy liści wewnątrz zamaskowanych obszarów punktów danych. *Filtrowanie Zbioru Danych* odrzuca punkty danych z wynikami wypełniania, które zmniejszają spójność z ich adnotacjami, przy użyciu modelu segmentacji semantycznej.

Model Kształtu i Tekstury Liścia: generowanie zakresu kształtów, rozmiarów i tekstur liści w celu zwiększenia zmienności zbioru danych; *(3) Oznaczanie Masek Semantycznych i Powierzchni:* Maski semantyczne wyznaczają granice liści na tle papieru milimetrowego, wraz z dokładnymi etykietami powierzchni; *(4) Renderowanie i Ostateczna Kompozycja Obrazu:* syntetyczne liście są łączone z tłem papieru milimetrowego, dodając realistyczne oświetlenie, efekty cieniowania i ogólną kompozycję obrazu; *(5) Wypełnianie Zbioru Danych:* Każdy obraz w zbiorze danych jest przetwarzany przy użyciu procesu ControlNet[97] do dokładnego wypełniania masek liści przy użyciu krawędzi Canny'ego i opisów tekstowych jako wejścia; *(6) Kontrola Jakości na Bazie Segmentacji Semantycznej:* Syntetyczne obrazy wypełnione przez ControlNet są segmentowane semantycznie na liście i tło, a następnie porównywane z proceduralnie wygenerowanymi maskami referencyjnymi w celu ustalenia spójności adnotacji.

3.2 Powiązane prace

Modele głębokiego uczenia wykazały duży postęp w wielu dziedzinach, ale wymagają dużej ilości wysokiej jakości danych do efektywnego treningu. Podczas gdy dane są obfite, dane oznakowane są kosztowne i trudne do zdobycia, szczególnie w naukach przyrodniczych, gdzie zmienność pojedynczego gatunku biologicznego może być znacznie wysoka zarówno pod względem kształtu, jak i wyglądu (tekstury). W celu sprostania temu wyzwaniu

opracowano różne podejścia, w tym uczenie półnadzorowane, samonadzorowane oraz generowanie danych syntetycznych.

Jednym z godnych uwagi podejść w generowaniu danych syntetycznych jest DatasetGAN [98], które zaproponowało proces generowania początkowych obrazów przez StyleGAN, a następnie ręczną anotację kilku obrazów dla konkretnego zadania, po czym trenowany jest mały model do produkcji podobnych masek segmentacyjnych z cech StyleGAN. Metoda ta pozwala na generowanie dużej liczby oznakowanych obrazów przy minimalnym wysiłku manualnym. BigDatasetGAN [98] rozszerzyło tę koncepcję, wykorzystując BigGAN do generowania szerokiej gamy obrazów, skalując ją do złożoności zbiorów danych, takich jak ImageNet.

Sun et al. [84] wprowadzili SHIFT, syntetyczny zbiór danych jazdy samochodem z wariacjami pogodowymi, porą dnia oraz gęstością pojazdów i pieszych, wykorzystując adaptację domeny do realistycznych symulacji. Yan et al. [94] opracowali system lokalizacji wizualnej wykorzystujący narzędzie do generowania danych syntetycznych, które łączy rzeczywiste i syntetyczne światy, generując dane z wieloma adnotacjami.

Podobnie, Anagnostopoulou et al. [1] opracowali realistyczny syntetyczny zbiór danych scen z grzybami do wykorzystania w robotyce zbioru grzybów. Bliskie naszemu podejściu są prace Ubbens et al. [87], które opracowały syntetyczny model liścia roślin rozetowych do zliczania, koncentrując się na opisie morfologii całej rośliny. Przedstawione podejście prezentuje ukierunkowaną metodę generowania dużego zbioru danych liści dla różnych gatunków drzew. Ponadto, Zhang et al. [97] zademonstrowali postępy w modelach dyfuzji tekst-do-obrazu poprzez dodanie warunków kontrolnych.

Wygląd i modelowanie liści są badane przez grafikę komputerową od dekad. Chiba et al. [14] zaproponowali metodę kolorowania i układu liści. Wang et al. wykorzystali fizykę do symulacji wzrostu liści w pracy [89], a wzory unerwienia liści były badane w [75, 34]. Ogólne podejście do rozwoju kształtu liścia uwzględniające dane eksperymentalne z biologii rozwojowej zaproponowano w [77].

Praca ta opiera się na istniejących badaniach, badając integrację wydajnych, kontrolowalnych 3D modeli proceduralnych podobnych do tych używanych w pracy Raistrick et al. [70] w procesie wykorzystującym modele SI generatywnej do trenowania modeli głębokiego uczenia dla specjalistycznych zadań wizji komputerowej, gdzie dane rzeczywiste są rzadkie lub kosztowne do zdobycia.

3.3 Metoda

LAESI umożliwia w pełni automatyczne, masowe generowanie syntetycznych danych, wykorzystując proste, ale wydajne obliczeniowo modele proceduralne 3D do renderowania. Proceduralne modele zostały zaimplementowane w Unity. W tej sekcji omówiono poszczególne komponenty procesu modelowania i renderowania LAESI.

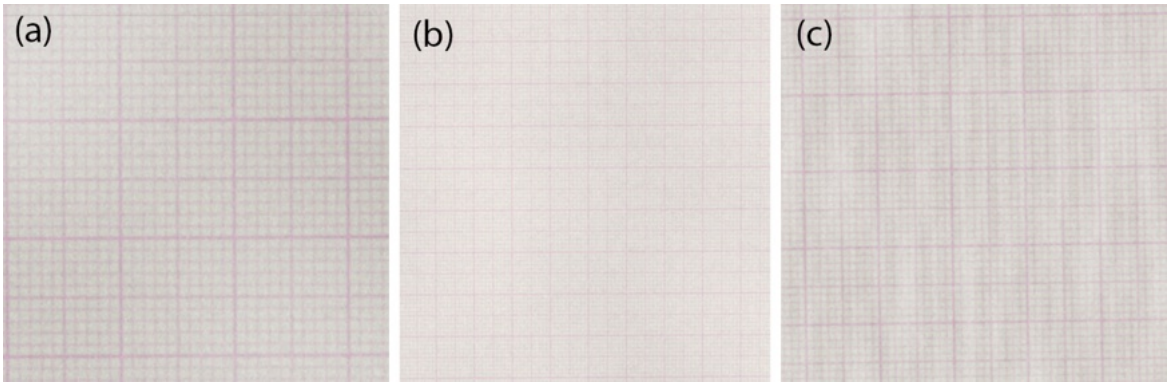

Fig. 3.2 Przykłady różnych tekstur papieru milimetrowego wygenerowanych za pomocą metody shaderów proceduralnych, od ostrych po rozmyte.

3.3.1 Proceduralny model papieru milimetrowego

Proceduralny model papieru milimetrowego generuje unikalne tekstury papieru milimetrowego, które służą jako tło w scenach renderowanych liści.

Metoda jest zaimplementowana w shaderze fragmentów poprzez proceduralne modelowanie tekstury standardowego papieru milimetrowego przy użyciu funkcji sinusoidalnych. Dla fragmentu o pozycji lokalnej (x,y) na papierze milimetrowym, intensywność koloru jest określona przez następującą funkcję:

$$C(x,y,\phi) = A \cdot \sin(B \cdot x + \phi) + D, \tag{3.1}$$

gdzie A to amplituda pasków, która kontroluje ich zmienność intensywności, B to ich częstotliwość, która określa odległość między nimi, ϕ to przesunięcie fazowe, które przesuwa je poziomo, a D to bazowa intensywność koloru.

Dodatkowe efekty shaderów obejmują zmiany odcienia, kontrastu, jasności i nasycenia, aby emulować różne warunki papieru. Następnie, tę teksturę łączę z wieloma warstwami szumu. Użyto trzech rodzajów szumu: gradientowego G, Voronoi V i prostego S. Wybór wyników renderowania stworzonych przez różne konfiguracje wartości parametrów jest pokazany na rysunku 3.2.

Kombinacja warstw szumu jest podana przez ważoną sumę:

$$L(x,y) = w_G \cdot G(x,y) + w_V \cdot V(x,y) + w_S \cdot S(x,y)\,,$$

gdzie w to wagi określające siłę szumu.

3.3.2 Proceduralny model liścia

Przedstawiony algorytm do proceduralnego generowania modeli liści obejmuje kilka etapów obliczeniowych, aby symulować morfologię liści. Początkowy kształt jest definiowany na CPU za pomocą krzywej animacji Unity, która jest parametryczną krzywą wielomianową definiowaną przez zestaw punktów kontrolnych interpolujących wartości, tworząc płynne przejścia.

Na pozycje punktów kontrolnych są nakładane losowe perturbacje, aby wprowadzić zmienność, odzwierciedlając inherentną różnorodność kształtów liści. Ta losowość jest matematycznie wyrażona przez dodanie funkcji szumu N do pozycji punktu kontrolnego P, gdzie nowa pozycja to $P' = P + N$.

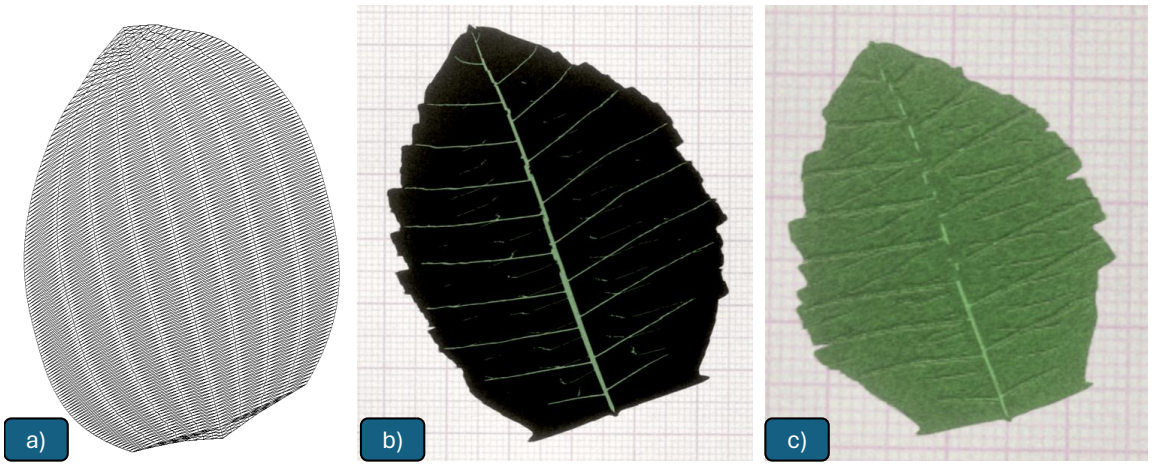

Fig. 3.3 Generowanie proceduralnego modelu liścia: Kształt jest definiowany przez parametryczną krzywą za pomocą krzywej animacji Unity (a), która następnie jest teksturowana, w tym rozwój wzoru unerwienia (b), a elementy stochastyczne i detale powierzchniowe są dodawane przez efekty shaderów (c).

Wzory unerwienia są generowane zgodnie z metodą przedstawioną przez Goldman et al. [23], wykorzystując system grafiki żółwia. System ten jest formalizowany jako seria komend definiujących ruch żółwia, gdzie ścieżka żółwia w przestrzeni odwzorowuje unerwienie - generowane w przestrzeni ekranu jako tekstura proceduralna. Dodawany jest element

stochastyczny, wprowadzając ruch Browna $Bm(t)$ do urozmaicenia ruchu żółwia, aby tworzyć faliste linie unerwienia i losowość kątów rozgałęzień θ_{veins} dla ścieżek unerwienia.

Generowanie tekstur obejmuje również tworzenie mapy wysokości H dla reprezentacji głębi, gdzie H jest modyfikowane przez funkcję ścieżki żółwia i odpowiadającą jej grubość unerwienia. Mapa wysokości jest używana do uzyskania normalnych dla mapowania normalnych w shaderze fragmentów w celu uzyskania szczegółowych tekstur powierzchni liścia (przykłady pokazane na rysunku 3.3).

Przemieszczenia wierzchołków w siatce są wprowadzane przy użyciu funkcji szumu Voronoi V w shaderze wierzchołków, aby symulować pofałdowania powierzchni liścia. Shader fragmentów następnie wykorzystuje mieszaninę szumów proceduralnych i fotorealistyczną teksturę liścia. Początkowo definiowane są dwie podstawowe tekstury kolorów, C_1 i C_2, reprezentujące różne aspekty koloryzacji i wzorów liścia. Ostateczna tekstura liścia T jest wynikiem mieszania tych tekstur kolorów, modulowanych przez szum gradientowy $G(x,y)$. Dodatkowo wprowadzone są szczegóły: unerwienie (oznaczone jako V_e), dziury (oznaczone jako H_o), oraz inne elementy teksturalne, takie jak plamy i nieregularności krawędzi, które naśladują naturalne niedoskonałości w morfologii liści.

3.4 Implementacja

Renderowanie Każdy liść przechodzi przez cztery oddzielne etapy renderowania, aby uchwycić różne wyglądy (przykłady pokazane na rysunku 3.4). W każdym etapie dostosowane są parametry shaderów do renderowania cieni, w tym siłę (s), pozycję ($\vec{p}$), i rozmiar (sz, odnoszący się do rozmiaru cienia), aby symulować różne warunki oświetleniowe. Parametry te są zarządzane przez funkcję cienia $S(s,\vec{p},sz)$, która jest zaimplementowana przy użyciu algorytmu mapowania cieni.

Skalowanie Tła Rozmiar papieru milimetrowego w scenie jest dostosowywany za pomocą współczynnika skalowania γ, zachowując spójną perspektywę kamery we wszystkich renderingach. Skalowane tło jest oznaczone jako $M' = \gamma M$.

Kompozycja Sceny Dodatkowe elementy, takie jak fragmenty papieru i szkło, są uwzględniane w celu zwiększenia realizmu. Ich transformacje w scenie są obsługiwane za pomocą kombinacji losowych translacji (T_{xy}), rotacji (R_θ), i skalowań (S_{xy}).

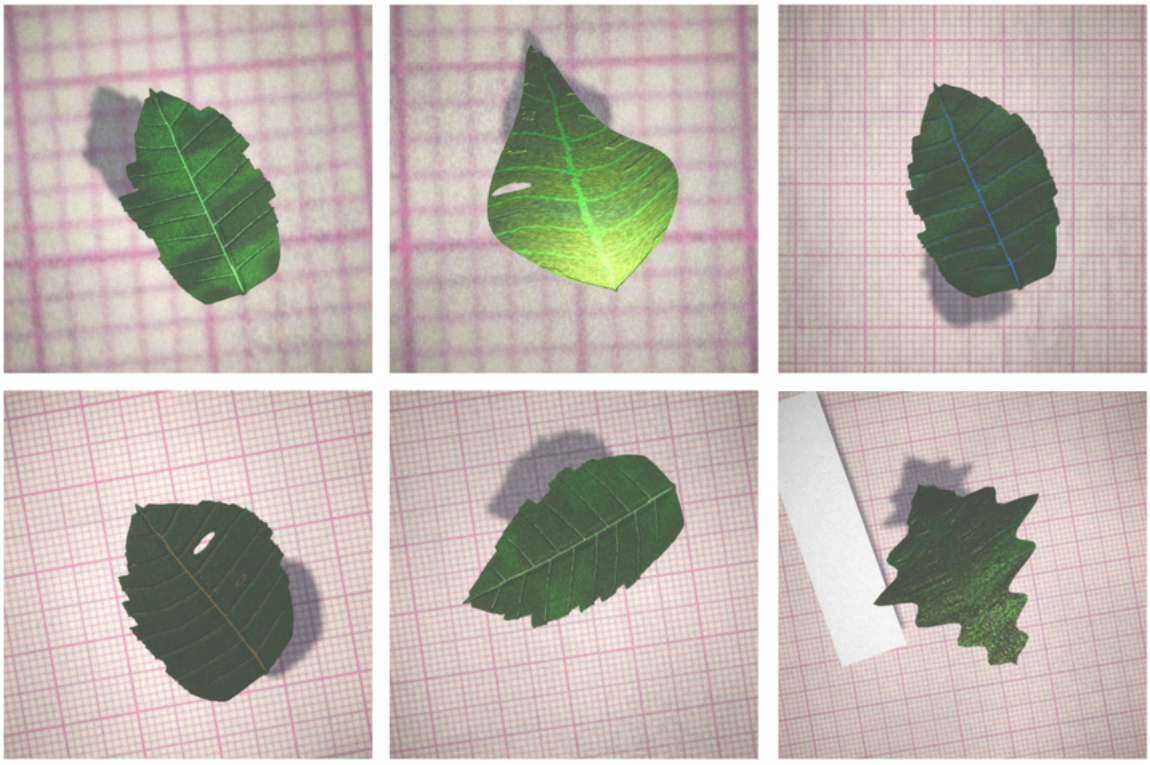

Fig. 3.4 Różnorodne renderingi końcowe z procesu generowania proceduralnego liści. Ta kolekcja ilustruje zmienność osiągniętą w wyglądzie liści dzięki parametrom przedstawionego modelu proceduralnego. Każdy rendering przedstawia różne warunki oświetleniowe, efekty cieniowania i skalowanie tła.

3.4.1 Przygotowanie danych

Ważnym aspektem tworzonego zbioru danych jest generowanie masek semantycznych i precyzyjne obliczanie etykiet powierzchni liści. Te elementy są kluczowe dla różnych zastosowań, w tym analizy morfologicznej i trenowania modeli ML.

Inicjowany jest kolejny etap procesu graficznego dla każdego renderowanego liścia z jednokolorową siatką liści na czarnym tle, aby uzyskać maskę semantyczną. Obliczenie rozmiaru powierzchni liścia jest bezpośrednim zastosowaniem parametrów proceduralnych używanych w niniejszym modelu papieru milimetrowego. Biorąc pod uwagę, że model papieru ma siatkę generowaną o znanych wymiarach, a współczynnik skalowania (γ) używany w renderowaniu jest również znany, mogę dokładnie obliczyć powierzchnię każdego liścia, sumując powierzchnie wszystkich trójkątów tworzących siatkę powierzchni liścia.

3.4.2 Integracja wypełniania ControlNet

Po wygenerowaniu początkowych 100 000 anotowanych obrazów syntetycznych wraz z ich maskami, użyto tych obrazów do uzyskania realistycznych 3D anotowanych odpowiedników. Aby osiągnąć ten cel, wykorzystano wytrenowaną sieć ControlNet zaproponowaną przez Zhang et al. [97].

W przedstwionej implementacji model Stable Diffusion [74] służy jako podstawa, z strukturą U-Net obejmującą enkoder, blok środkowy i dekoder z połączeniami przeskakującymi.

Model wykorzystuje konwolucje do down-samplingu i up-samplingu, bloki ResNet oraz warstwy Vision Transformer do ekstrakcji cech. Wejścia tekstowe są kodowane przy użyciu modelu CLIP, co jest konieczne do procesu generowania w kierunku specyficznych atrybutów opisanych tekstem [68].

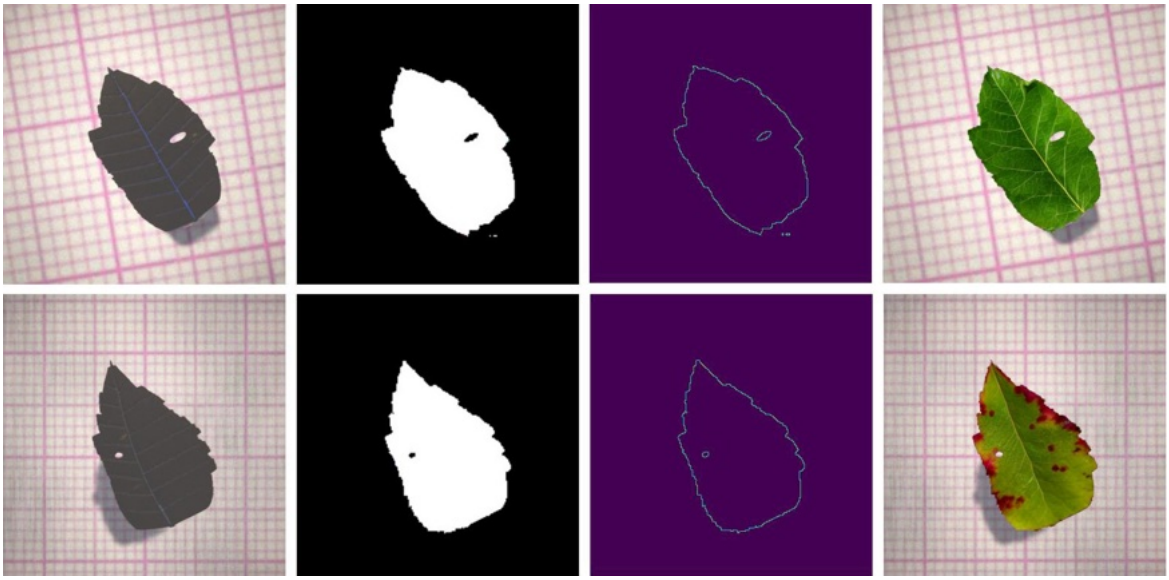

Fig. 3.5 Przykład wyników wypełniania przy użyciu ControlNet do wypełniania masek semantycznych. W dolnym rzędzie proces wypełniania dodał cechy chorobowe, które nie były opisane w modelu proceduralnym i byłyby bardzo trudne do proceduralnego zasymulowania.

ControlNet oferuje szereg wytrenowanych sieci z różnymi warunkami obrazowymi do regulacji modeli dyfuzji. Te warunki obejmują krawędzie generowane różnymi metodami, mapy głębokości i normalne, pozy człowieka, segmentację semantyczną, szkice użytkownika, itp. W przedstawionych eksperymentach zastosowano metodę wykrywania krawędzi Canny'ego [11]. Specyficznie, aby generować realistyczne obrazy z każdego syntetycznego obrazu, tworzę obraz wykorzystując krawędzie Canny'ego naszych syntetycznych masek jako wejścia. Tekstowe wejście składa się ze specyficznych fraz „liść

dębu na papierze milimetrowym" i „liść buka na papierze milimetrowym", które dostarczają danych wejściowych do ControlNet w celu wypełnienia pożądanych atrybutów obrazów syntetycznych liści. Następnie zastąpino tło naszym proceduralnie wygenerowanym. Należy zauważyć, że nie można użyć tła papieru milimetrowego z obrazu wygenerowanego przez AI, ponieważ wówczas etykiety powierzchni liści stałyby się niespójne.

Fig. 3.6 Trzy przykłady obrazów wygenerowanych przez ControlNet, gdzie obszar wypełnionego liścia w masce znacząco odbiega od obszaru zdefiniowanego przez maskę proceduralnie wygenerowaną. Takie punkty danych są automatycznie filtrowane w *LAESI*.

Po generacji przy użyciu ControlNet, dane syntetyczne przechodzą proces filtrowania w celu eliminacji obrazów niespójnych z przypisanymi im adnotacjami. Proces ten wykorzystuje model segmentacji semantycznej oparty na MobileNet, wytrenowany na zbiorze danych *Synthetic Rendering 2*, obejmującym 5000 obrazów syntetycznych i 1700 rzeczywistych (patrz Tab. 3.1). Syntetyczne obrazy, w których przewidywana maska odbiega o więcej niż 15% od prawdziwej maski, jak w przypadkach pokazanych na rysunku 3.6, są usuwane ze zbioru danych. Ten krok usuwa outliery, które zostały napotkane podczas etapu wypełniania ControlNet z częstotliwością 15-20%.

3.5 Walidacja

W celu przewidywania rozmiaru powierzchni liści oraz segmentacji semantycznej rzeczywistych liści na papierze milimetrowym, zastosowano dane syntetyczne do trenowania modeli. Zadanie to staje się zaskakująco trudne z powodu używania różnych sensorów, parametrów zewnętrznych kamery oraz innych artefaktów obrazu, takich jak refleksy, notatki i obiekty pojawiające się na fotografiach w rzeczywistym środowisku badawczym. Zastosowanie metod opartych na regułach do przewidywania powierzchni liści zazwyczaj wymaga silnie kontrolowanych środowisk [69] lub specyficznych obiektów referencyjnych włączonych w celu skalowania [17], co czyni te metody niepraktycznymi dla typowych prac badawczych.

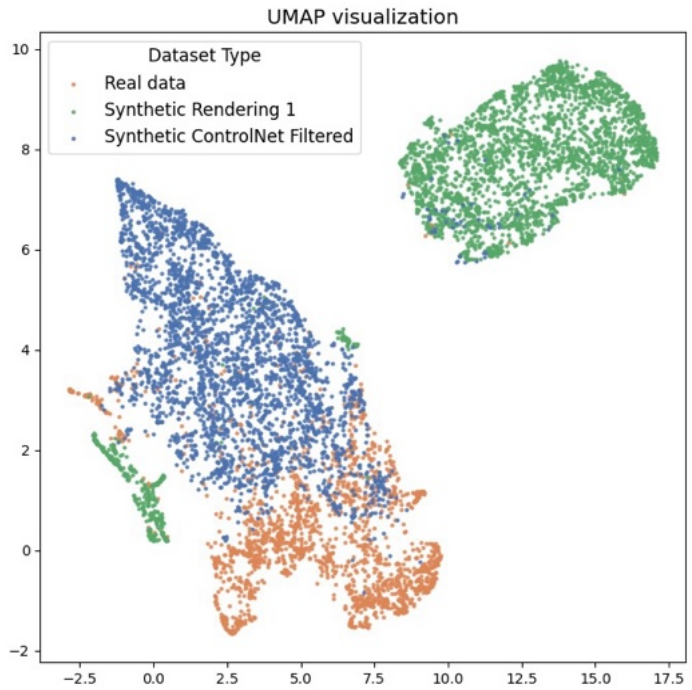

Fig. 3.7 Wizualizacja UMAP w przestrzeni embeddingów ResNet50 (CLIP ViT-B/32) dla danych rzeczywistych (pomarańczowy) i dwóch różnych zestawów obrazów syntetycznych (*Rendering 1* - zielony, *ControlNet + Filtering* - niebieski). Brak separacji w przestrzeni cech między niebieskimi a pomarańczowymi kropkami sugeruje, że obrazy syntetyczne w zestawie *ControlNet + Filtering* zawierają cechy semantycznie bardziej podobne do rzeczywistych w porównaniu do zestawu Rendering 1.

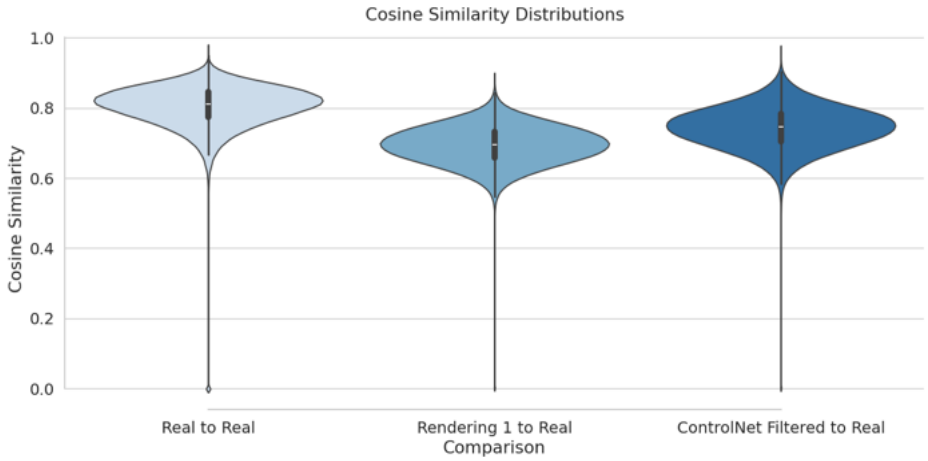

Fig. 3.8 Wykresy skrzypcowe wyników podobieństwa cosinusowego dla zbiorów danych użytych w Fig. 3.7. Rozkład obrazów *ControlNet Filtered* ma ogólnie wyższe wyniki podobieństwa cosinusowego w porównaniu do zbioru *Rendering 1*. Rozkłady pochodzą z obrazów, które mają ogólnie podobne cechy w porównaniu do rzeczywistych, co wskazują wysokie wyniki.

3.5.1 Trenowanie modelu sieciowego

Użyto MobileNet V3 [35] do przewidywania powierzchni liści poprzez regresję. Wybrano MobileNet V3, który został zaprojektowany do aplikacji mobilnych działających na urządzeniach przenośnych, aby ułatwić wdrożenie tego rozwiązania w zdalnych lokalizacjach.

3.5.2 Optymalizacja hiperparametrów

W trakcie eksperymentów systematycznie zmieniono kilka hiperparametrów, w tym (1) Wariacje architektury (MobileNet V3 Large i Small). (2) Techniki augmentacji danych (jasność, kontrast, odcień, nasycenie, odwrócenie, obrót o 90^o, losowy obrót, szum). Z powodu braku identyfikacji optymalnej konfiguracji hiperparametrów w literaturze naukowej, zastosowano szeroką eksplorację przestrzeni hiperparametrów, przeprowadzając 1425 eksperymentów w celu zidentyfikowania najbardziej wydajnych modeli poprzez crowdsourcing.

Optymalne hiperparametry zostały zidentyfikowane na podstawie wydajności na zbiorze walidacyjnym. Najlepsze ustawienia obejmowały MobileNet V3 Large z transfer learningiem ImageNet (wstępnie wytrenowane z wagami ImageNet-1k), optymalizator RMSprop, początkową szybkość uczenia $1e^{-3}$ ze zmiennym spadkiem do $1e^{-9}$ i augmentacje obejmujące jasność, kontrast, odcień, nasycenie, odwrócenie, obrót o 90^o i szum Poissona.

3.5.3 Eksperymenty Walidacyjne

Przeprowadzono sześć eksperymentów treningowych:

1. *Baza oparta na regułach*: 30 rzeczywistych punktów danych z jednolicie wielkimi czerwonymi kwadratami dla celów kalibracyjnych na obrazach. Wydajność oceniono za pomocą metody „Easy Leaf Area” do przewidywania powierzchni liści opartej na regułach [17], w przeciwieństwie do innych eksperymentów, które wykorzystują MobileNet V3.

2. *Baza Danych Rzeczywistych*: Trening z 1,7 tys. rzeczywistych, oznakowanych punktów danych liści (przykłady pokazane na rysunku 3.9).

3. *Rendering Syntetyczny 1*: Trening z 1,7 tys. rzeczywistych punktów danych w połączeniu z 5 tys. punktów danych syntetycznych.

4. *Rendering Syntetyczny 2*: Trening z 1,7 tys. rzeczywistych punktów danych w połączeniu z 5 tys. punktów danych syntetycznych przy użyciu poprawionych

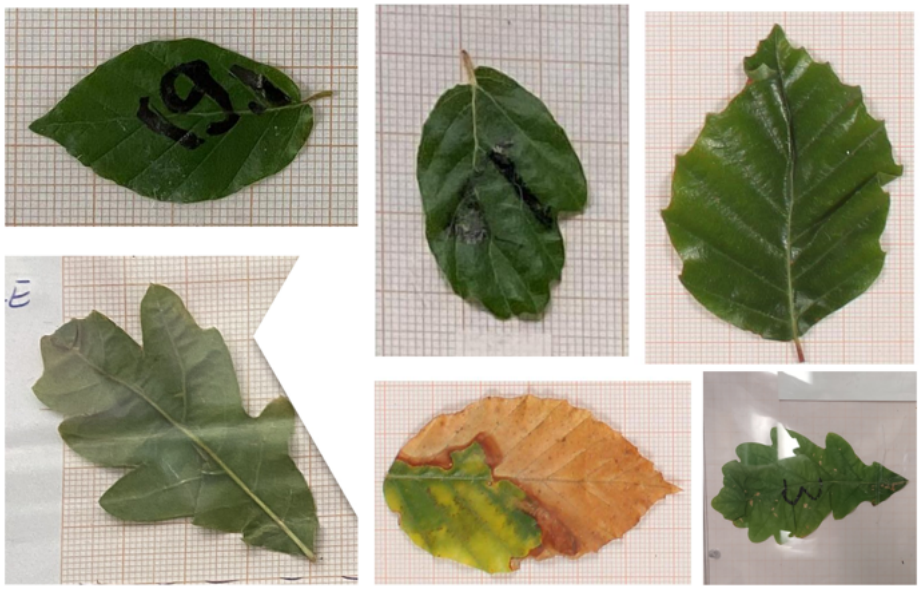

Fig. 3.9 Fotografie liści buka i dębu na papierze milimetrowym wykonane w TUM School of Life Sciences. Zostały one użyte w celu stworzenia bazy do trenowania modeli sieciowych do przewidywania powierzchni liści oraz segmentacji semantycznej, a także do zbioru danych walidacyjnych.

konfiguracji wartości parametrów opartych na wynikach uzyskanych z poprzedniego zestawu danych. Zmieniono wartości parametrów dla modelu papieru milimetrowego, aby usunąć szum papierowy, zawężając zakres wartości parametrów funkcji szumu.

5. *ControlNet Syntetyczny*: Trening z 1,7 tys. rzeczywistych punktów danych w połączeniu z 5 tys., 6 tys., 7 tys., 8 tys., 9 tys. i 10 tys. punktów danych syntetycznych, wykorzystując ControlNet do wypełniania masek.

6. *ControlNet Syntetyczny + Filtrowanie*: Trening z użyciem ControlNet i procesu filtrowania 5 tys., 6 tys., 7 tys., 8 tys., 9 tys. i 10 tys. punktów danych syntetycznych, zmieszanych z 1,7 tys. rzeczywistych punktów danych.

Każdy eksperyment wykorzystał te same zbiory danych walidacyjnych i testowych, obejmujące 250 rzeczywistych zdjęć z adnotacjami. Dane walidacyjne zostały zebrane z eksperymentów w komorze klimatycznej przeprowadzonych w TUMmesa ecotron oraz na Uniwersytecie Technicznym w Monachium (TUM) w latach 2021 i 2022. Rzeczywiste adnotacje uzyskano empirycznie i przy użyciu miernika powierzchni LICOR LI-3100C.

3.5.4 Metryki ewaluacji

Podstawową metryką oceny jest średni błąd względny (MRE) powierzchni liści między przewidywanymi a rzeczywistymi wartościami. Dla segmentacji semantycznej zastosowano metrykę średniego współczynnika przecinania (mIoU) między maskami rzeczywistymi a przewidywanymi, a także względny błąd w całkowitej liczbie pikseli maski - nazywany błędem liczby pikseli maski (MPE). Dodatkowo, zastosowano wyniki podobieństwa cosinusowego oraz metodę redukcji wymiarów UMAP [49] do ilościowego określenia podobieństwa między obrazami syntetycznymi a rzeczywistymi (Fig. 3.10- 3.8).

Fig. 3.10 Dwie pary syntetycznych (po lewej) i rzeczywistych (po prawej) obrazów wybranych z 100 najwyższych wyników podobieństwa cosinusowego z zestawu danych *Rendering 2* i poniżej z zestawu *ControlNet+Filtering*.

Wyniki eksperymentów z treningiem syntetycznym, podsumowane w tabeli 3.1, ujawniają znaczące różnice w wydajności modelu w zależności od rodzaju i ilości danych syntetycznych użytych do trenowania modeli sieciowych. Eksperyment *Baza Danych Rzeczywistych* osiągnął MRE powyżej 12,5% na zestawach walidacyjnych i testowych. Eksperymenty *Rendering 1+2* z mieszanymi danymi syntetycznymi wykazały niewielką poprawę MRE walidacyjnego. Jednak najbardziej znaczące postępy zaobserwowano w eksperymentach *ControlNet Syntetyczny* i *ControlNet Syntetyczny + Filtrowanie*. Tutaj włączenie 10 tys. punktów danych syntetycznych z filtrowaniem znacznie obniżyło MRE do poziomu 6,1% na walidacji i 6,2% na testach (rysunek 3.11). Co ciekawe, zwiększenie liczby punktów

danych skutkuje proporcjonalnie większą poprawą MRE dla eksperymentu ControlNet + Filtrowanie w porównaniu do innych eksperymentów. Ponadto, chociaż MRE dla przewidywania powierzchni liści poprawiło się w przypadku danych wypełnionych (z filtrowaniem i bez), wydajność mierzona przez mIoU i MPE dla modelu segmentacji semantycznej zmniejszyła się w eksperymencie ControlNet (0,09 MPE) w porównaniu do eksperymentów bez wypełnienia (0,07 MPE), ale wzrosła w eksperymencie ControlNet + Filtrowanie (0,05 MPE, Tab. 3.1). Pokazuje to użyteczność filtrowania po etapie wypełniania, co znacznie poprawia wydajność w obu zadaniach wizji komputerowej.

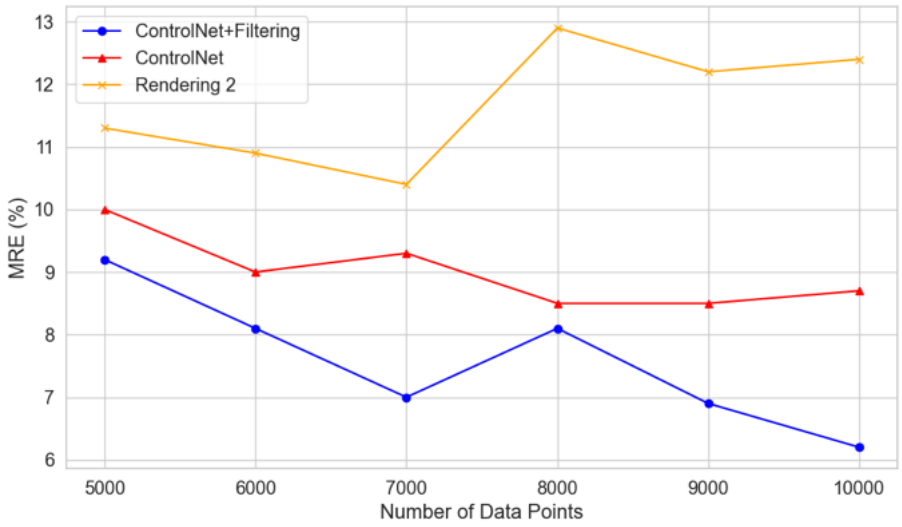

Fig. 3.11 Krzywe utraty danych walidacyjnych dla eksperymentów z treningiem na zestawach danych od 5 tys. do 10 tys. punktów. Czerwona krzywa wskazuje wyniki uzyskane z surowych danych wypełnionych ControlNet, niebieska krzywa z filtrowanymi danymi, a pomarańczowa bez wypełniania (Rendering 2). Podczas gdy dodanie większej ilości danych znacznie poprawia MRE przy przewidywaniu powierzchni liści na danych wypełnionych, nie ma poprawy dla surowych danych syntetycznych.

3.6 Dyskusja i wnioski

W niniejszym badaniu przedstawiono w pełni automatyczne podejście do generowania syntetycznych danych do przewidywania powierzchni liści, wykorzystując modele proceduralne i architekturę sieci neuronowej MobileNet V3. Podczas gdy standardowe metody wizji komputerowej do przewidywania powierzchni liści (np. [17]) mogą dobrze działać w silnie kontrolowanych kontekstach, okazało się, że typowe fotografie zebrane z eksperymentów laboratoryjnych zawierają nieoczekiwaną złożoność, co czyni te metody

Fig. 3.12 Wybór syntetycznych obrazów wygenerowanych za pomocą *LAESI*. Te obrazy są częścią podzbioru *ControlNet + Filtrowanie*.

Tabela 3.1 Porównanie wydajności eksperymentów treningowych z rzeczywistą bazą danych (1,7 tys.) i 5 tys. danych treningowych syntetycznych oraz bazą opartą na regułach.

Eksperyment	**Walidacja MRE** (%)	**Test MRE** (%)	**mIoU (%)**	**Błąd Liczby Pikseli Maski** (%)
Baza Oparta na Regułach	38.3	38.3	-	-
Baza Danych Rzeczywistych	12.5	12.9	0.79	0.08
Rendering 1	12.0	11.0	0.81	0.07
Rendering 2	10.8	11.3	0.82	0.07
ControlNet	8.4	10.0	0.8	0.09
ControlNet + Filtrowanie	8.5	9.2	0.83	0.05

trudnymi do użycia w praktyce. Obejmuje to notatki, refleksy, zmienne cechy wewnętrzne i zewnętrzne sensorów, kawałki papieru i inne obiekty obecne na obrazach (patrz rysunek 3.5). Dodatkowo, wyniki wskazują, że wypełnianie danych syntetycznych za pomocą ControlNet jest wykonalnym sposobem na poprawę wydajności w specjalistycznym zadaniu wizji komputerowej oraz że dodanie filtrowania do procesu generowania danych na podstawie spójności adnotacji znacznie poprawia ogólną wydajność. Wyniki wskazują, że dane syntetyczne przed wypełnieniem prawdopodobnie nie mają wystarczającej zmienności cech, aby przynieść korzyści w trenowaniu modeli na zwiększonej ilości punktów danych (rysunek 3.11, pomarańczowa krzywa), podczas gdy nieprzefiltrowane dane wypełnione zmniejszają wydajność modelu w zadaniu segmentacji semantycznej poprzez wprowadzanie niespójności do adnotacji (Tab. 3.1, wyniki mIoU i MPE). Zaskakująco, oznacza to, że trening na błędnych adnotacjach w eksperymencie ControlNet nadal prowadzi do lepszego ogólnego MRE w przewidywaniu powierzchni liści. Przedstawione analizy wskazują, że najlepsza adaptacja domeny dla proceduralnie generowanych danych syntetycznych może być osiągnięta przez filtrowanie wypełniania, co redukuje niespójności adnotacji przy jednoczesnym zachowaniu cech generowanych przez AI. Wnioski te są podobne do tych przedstawionych przez Fei et al. [22], którzy pokazali, że zastosowanie straty ograniczającej semantykę w treningu metod opartych na GAN może pomóc w utrzymaniu spójności adnotacji, co poprawia wydajność modelu.

Poprzez serię eksperymentów, oceniono różne konfiguracje wartości hiperparametrów i zbiorów danych, prowadząc do optymalnego zestawu, który osiągnął stratę testową na poziomie 6,2% w ostatecznym eksperymencie, co jest co najmniej na poziomie błędu ludzkiego oznaczania i znacznie przewyższa najlepszą rzeczywistą bazę danych. Doprowadziło to do przyjęcia tego modelu przez badaczy biologii w TUM do dalszych eksperymentów, co czyni ręczne oznaczanie zbędnym. Ręczne oznaczanie ponad 2 tys. obrazów liści to kosztowne i czasochłonne przedsięwzięcie, które wydłuża zbieranie wyników

empirycznych. Wyniki potwierdzają wartość danych syntetycznych w trenowaniu modeli głębokiego uczenia do specyficznej aplikacji badawczej, wzmacniając tezy o przydatności danych syntetycznych przedstawione w innych pracach badawczych (np. [20, 80, 95, 40]).

Ponadto, podejście, które łączy generowanie proceduralne z metodami SI generatywnej, posiada potencjał do różnych innych zastosowań w badaniach botanicznych i rolnictwie, szczególnie w teledetekcji i rolnictwie precyzyjnym. Podejście to można rozszerzyć na inne dziedziny, co pozwoli na w pełni automatyczne generowanie syntetycznych danych do uczenia maszynowego.

Rozdział 4

Tworzenie mapy głębokości dla wielu sadzonek na pojedynczym zdjęciu

4.1 Wprowadzenie

Celem badań przedstawionych w tym rozdziale jest przewidywanie map głębokości dla sadzonek drzew w lizymetrze, bazując na analizie obrazów 2D. W przeciwieństwie do rozdziału 2, gdzie celem jest rekonstrukcja całych drzew na podstawie pojedynczych obrazów, w tym rozdziale wykorzystano zaawansowane techniki tworzenia danych syntetycznych, aby precyzyjnie odtworzyć strukturę sadzonek w skali centymetrowej. Dokładność tych modeli ma kluczowe znaczenie, gdyż pozwala na precyzyjne obliczanie powierzchni liści, co z kolei umożliwia szczegółowe pomiary wzrostu roślin oraz oceny wpływu środowiska na ich zdrowie.

W rozdziale 3 przedstawiono metodologię generowania syntetycznych danych morfologii liści przy użyciu modeli proceduralnych i systemu ControlNet. Niniejszy rozdział kontynuuje tę pracę, rozszerzając ją na przypadki analizy sadzonek drzew, co adresuje trudności, takie jak przesłonięcia liści, złożoność wzorców rozgałęzień i zachowanie precyzyjnej skali. Na jednym zdjęciu istnieje wiele elementów, które są brane pod uwagę (wiele sadzonek, a każda ma wiele liści).

Kluczowym aspektem tego rozdziału nie jest opracowanie nowych architektur sieci neuronowych, lecz stworzenie wysokiej jakości syntetycznych danych, które umożliwiają trenowanie modeli o wysokiej dokładności. W tym celu wykorzystano dane syntetyczne o rozdzielczości 512x512 pikseli, generowane przy użyciu modeli proceduralnych, takich jak te opisane w rozdziale 2 oraz mechanizmów ControlNet (podobnie jak w rozdziale 3) w celu zapewnienia poprawnej skali i struktury obrazów.

Jednym z głównych wyzwań, jest precyzyjne przewidywanie głębokości, co ma kluczowe znaczenie dla odtworzenia rzeczywistej struktury sadzonek w trójwymiarze. Potencjalnie umożliwia to pomiar wzrostu. W przedstawionej metodologii wykorzystano architekture U-Net do przewidywania głębokości. Dodatkowo wykorzystano modele state of the art, które dostrajono do zadanego problemu.

Dzięki dokładnemu przewidywaniu map głębokości uzyskano szczegółową i realistyczną reprezentację sadzonek, co umożliwia obliczanie powierzchni liści z wysoką precyzją. Jest to niezbędne do oceny wzrostu roślin oraz monitorowania wpływu środowiska. Podejście to wspierają prace, takie jak [10], które podkreślają skuteczność używania map głębokości do zadań rekonstrukcji 3D.

W kolejnych sekcjach tego rozdziału przedstawone jest porównanie wyników modeli trenowanych na syntetycznych danych, które zostały wygenerowane w oparciu o procedury omówione w rozdziałach 2 i 3. Analiza obejmuje wyniki uzyskane na zbiorze treningowym oraz walidacyjnym, z wykorzystaniem miar takich jak Średni Błąd Kwadratowy (MSE), Średni Błąd Bezwzględny (MAE) oraz ich odmiany maskowane, uwzględniające jedynie piksele znajdujące się wewnątrz lizymetru.

4.2 Przegląd literatury

W literaturze dotyczącej przewidywania map głębokości oraz wykorzystania syntetycznych danych w widzeniu komputerowym istnieje wiele badań, które przyczyniły się do rozwoju tej dziedziny. Proceduralne generowanie danych syntetycznych, które jest opisane w rozdziale 3, znajduje swoje korzenie w pracach takich jak [1], gdzie modele proceduralne zostały wykorzystane do tworzenia realistycznych scen syntetycznych, umożliwiających trenowanie modeli głębokiego uczenia. Takie podejście jest szczególnie wartościowe, gdy dostęp do rzeczywistych danych jest ograniczony lub ich pozyskanie jest kosztowne.

ControlNet, jako narzędzie do precyzyjnego kierowania procesem generowania obrazów syntetycznych, opisywane jest w [96], gdzie jego zastosowanie w kombinacji z mapami głębokości pokazuje efektywność w zachowaniu spójności przestrzennej. Ten aspekt jest kluczowy w kontekście moich badań, gdzie precyzyjna rekonstrukcja 3D sadzonek wymaga zachowania dokładnych proporcji i odległości w trójwymiarowej przestrzeni.

Jednym z kluczowych narzędzi w przewidywaniu map głębokości jest model Depth Anything, który pozwala na generowanie szczegółowych map głębokości z pojedynczych obrazów. Wersja tego modelu, dostosowana do skali metrycznej, jest szczególnie istotna w kontekście moich badań nad sadzonkami drzew, gdzie precyzja odległości ma kluczowe znaczenie [5]. Dodatkowo, model U-Net, który zastosowałem w moim badaniu, zyskał

popularność w dziedzinie widzenia komputerowego dzięki swojej zdolności do efektywnego przetwarzania obrazów w kontekście tłumaczenia obraz-obrazu, co było istotne także w tym zadaniu przewidywania map głębokości [18].

Integracja danych syntetycznych z rzeczywistymi obrazami, wsparta przez techniki takie jak Low-Rank Adaptation (LoRA) [36], umożliwia efektywne dostrajanie modeli do nowych zadań, co jest istotne w kontekście moich eksperymentów, gdzie syntetyczne dane wspomagają proces trenowania modeli na rzeczywistych obrazach sadzonek.

Niniejszy rozdział czerpie z tych prac, rozwijając techniki generowania syntetycznych danych i stosując je w kontekście przewidywania map głębokości roślin w lizymetrze.

4.3 Metoda

Niniejsze podejście opiera się na wykorzystaniu syntetycznych danych do trenowania modeli głębokiego uczenia w zadaniach przewidywania map głębokości . Proces generowania syntetycznych obrazów jest zbliżonny do tego w rozdziale 3, jednak dodałem kilka istotnych zmian w celu poprawy jakości wyniku, przybliżenia dziedzinowego i dostosowania metody do obecego problemu, jakim są mapy głębokości.

4.3.1 Generowanie danych syntetycznych

Aby efektywnie generować duży zbiór danych potrzebnych do trenowania modeli przewidywania map głębokości, opracowany został program w środowisku Unity, wykorzystujący język C# oraz Universal Render Pipeline. Program ten umożliwia proceduralne generowanie drzew w oparciu o L-Systemy, co pozwala na tworzenie złożonych, realistycznych struktur roślinnych, które są następnie używane do generowania syntetycznych obrazów i map głębokości 2.5.

L-Systemy (Lindenmayer Systems) stanowią podstawę do generowania struktury drzew. Wykorzystano zestaw reguł produkcji, które determinują kształt drzewa, uwzględniając elementy losowe, co zapewnia różnorodność generowanych struktur. Reguły te opisują sposób rozgałęziania się drzewa oraz rozmieszczenia liści, uwzględniając losowość w parametrach takich jak kąty odchyleń i długości gałęzi, co prowadzi do tworzenia unikalnych struktur drzew przy każdym uruchomieniu procesu. Dodatkowo, aby uniknąć nachodzenia się liści, zaimplementowano sprawdzanie kolizji, co zapewnia realistyczne rozmieszczenie liści i gałęzi.

W Unity generowane są zarówno mapy głębokości, jak i maski segmentacyjne. Mapy głębokości umożliwiają precyzyjne odwzorowanie struktury drzewa w skali metrycznej,

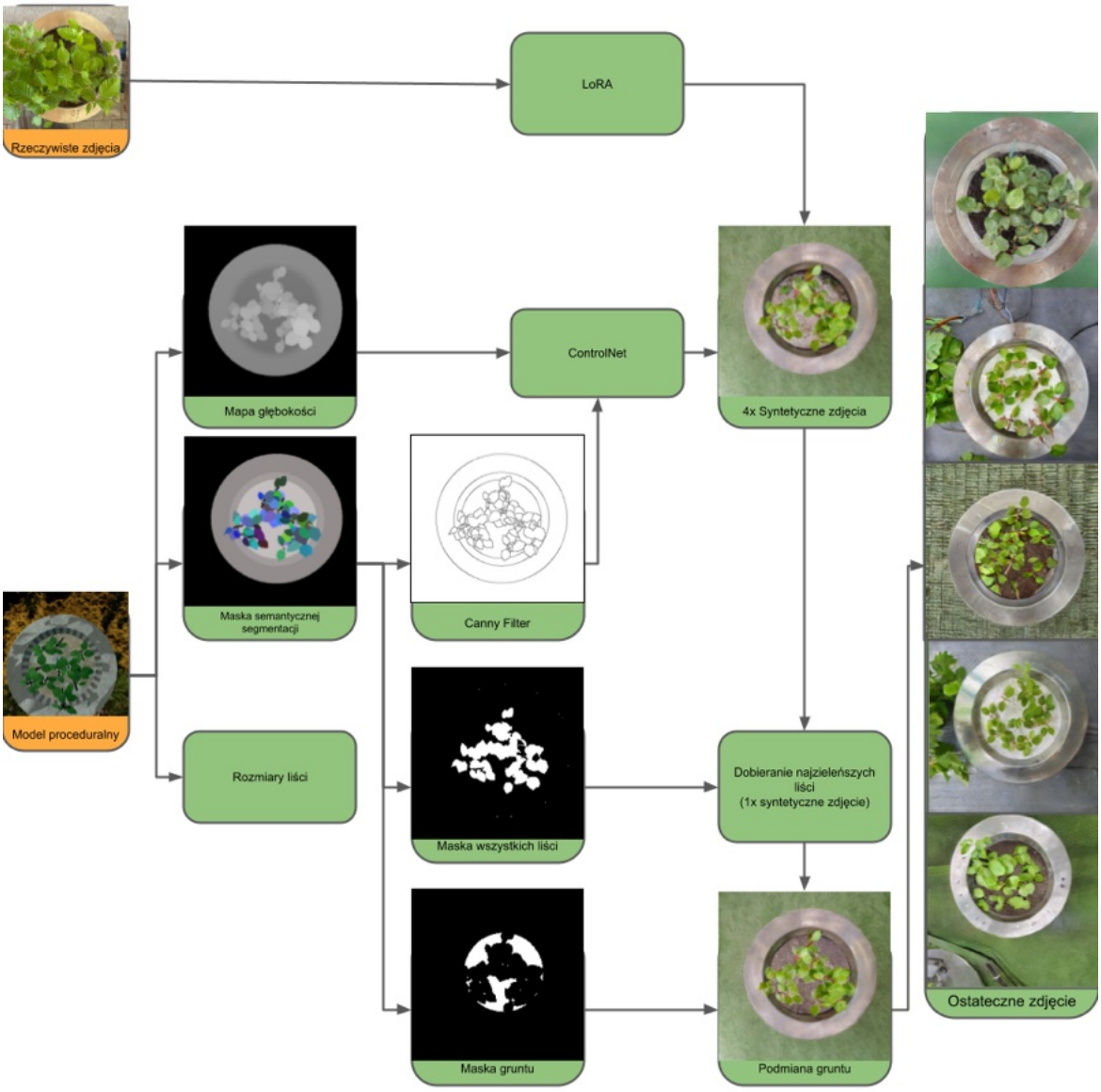

Fig. 4.1 Schemat generowania obrazów wraz z opisami przez proceduralny model w połączeniu z ControlNet'em, filtrem Canny oraz dostosowaną LoRA.

co jest kluczowe dla analizy morfologicznej roślin. Maski segmentacyjne natomiast służą do precyzyjnego rozdzielenia poszczególnych części drzewa, co jest istotne w kontekście dalszych procesów przetwarzania obrazów, takich jak wyodrembnianie pojedynczych organów i ich dokładniejsza analiza.

Na podstawie wygenerowanych w Unity map głębokości i masek segmentacyjnych, tworzone są obrazy do zbioru danych przy użyciu odpowiednich modeli ControlNet. Pierwszy ControlNet wykorzystuje mapy głębokości i maski segmentacyjne do generowania obrazów 2D, które wiernie odwzorowują strukturę drzewa, zachowując odpowiednią skalę i szczegóły.

Dodatkowo, drugi ControlNet przetwarza maski segmentacyjne, stosując filtr Canny, który wykrywa krawędzie w obrazie, a następnie generuje szczegółowe obrazy syntetyczne na podstawie tych krawędzi. Filtr Canny jest szczególnie przydatny w zadaniach, które wymagają precyzyjnego odwzorowania konturów i granic między różnymi elementami roślinnymi.

Ważnym elementem przedstawionego podejścia jest integracja modelu LoRA (Low-Rank Adaptation), który został wytrenowany na rzeczywistych zdjęciach sadzonek w lizymetrze (80 zdjęć). LoRA umożliwia efektywne dostrajanie modeli ControlNet do specyficznych zadań, co pozwala na poprawę dokładności generowanych obrazów syntetycznych. Dzięki zastosowaniu LoRA, modele te lepiej odwzorowują rzeczywiste warunki panujące w lizymetrze. Bardziej wiarygodnie przedstawiają zadane obiekty i dopasowują cechy charakterystyczne dla tych obiektów, np. cechy charakterystyczne dla gatunku. Przekłada się to na wyższą jakość danych treningowych. Pokazuje to jak ważne są dane rzeczywiste, nawet przy tworzeniu danych syntetycznych. Nawet jest ich nie wiele, to i tak mogą pozytywnie wpłynąć na proces generowania danych syntetycznych

Jednym z kluczowych aspektów tego podejścia jest zastosowanie elementów losowych w L-Systemach, co pozwala na tworzenie różnorodnych drzew przy każdym uruchomieniu programu. Dzięki temu możliwe jest wygenerowanie bogatego zbioru danych, który może być używany do trenowania modeli głębokiego uczenia w sposób bardziej wszechstronny i odporny na przetrenowanie. Losowość wprowadzana jest również na poziomie kątów odchyleń gałęzi, rozmiarów liści oraz ogólnej struktury drzewa.

Dzięki tak zorganizowanemu procesowi generowania danych, możliwe jest efektywne tworzenie wysokiej jakości danych syntetycznych, które pozwalają na dokładne przewidywanie map głębokości, co jest kluczowe w analizach morfologicznych roślin.

4.3.2 Szkolenie sieci

Model U-Net został wybrany ze względu na jego skuteczność w zadaniach związanych z tłumaczeniem obraz-obrazu [18]. Architektura modelu składa się z sieci konwolucyjnych,

które kolejno przetwarzają obrazy na coraz wyższych poziomach abstrakcji, a następnie dokonują rekonstrukcji mapy głębokości.

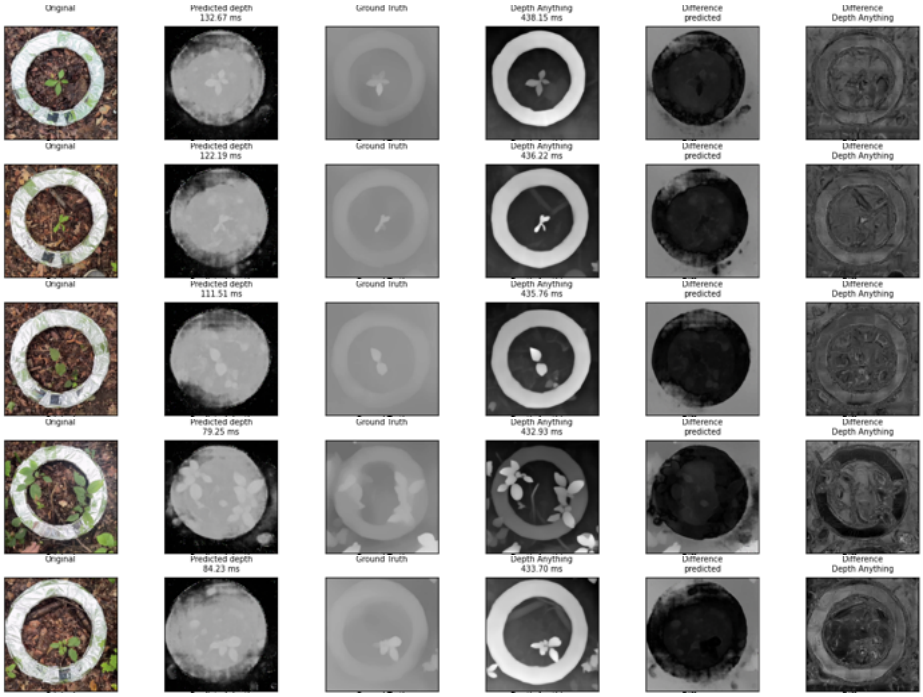

Fig. 4.2 Porównanie wyników estymacji głębokości za pomocą modeli U-Net i Depth Anything. Pierwsza kolumna przedstawia oryginalny obraz. Druga kolumna prezentuje przewidywaną mapę głębokości wygenerowaną przez model U-Net dostosowany do zadań tłumaczenia obraz-obraz. Trzecia kolumna pokazuje mapę głębokości uzyskaną za pomocą czujnika LiDAR iPhone, skalibrowaną w celu uchwycenia rzeczywistej głębokości w centymetrach. Czwarta kolumna przedstawia mapę głębokości wygenerowaną przez model Depth Anything skonfigurowany do przewidywań głębokości w jednostkach metrycznych. Piąta i szósta kolumna ukazują różnicę bezwzględną między prawdziwą mapą głębokości a przewidywaniami modeli U-Net i Depth Anything, odpowiednio podkreślając obszary, gdzie przewidywania odbiegają od rzeczywistych pomiarów głębokości. Widać, że wewnątrz lizymetru różnice są mniejsze przy użyciu tej metody w porównaniu do modelu Depth Anything.

Ważnym elementem procesu treningowego jest wykorzystanie metryk, które oceniają jakość przewidywań. W eksperymentach posłużono się Średnim Błędem Kwadratowym (MSE) oraz Średnim Błędem Bezwzględnym (MAE) jako głównymi miarami. Dodatkowo, wprowadzono metryki maskowane, które biorą pod uwagę jedynie piksele znajdujące się wewnątrz lizymetru. Pozwala to na dokładniejszą ocenę przewidywań w kontekście badanej struktury.

Drugim kluczowym modelem używanym w przedstawionych eksperymentach jest Depth Anything. Ten model został pierwotnie zaprojektowany do generowania szczegółowych map głębokości z pojedynczych obrazów przedstawiających dowolne sceny, przedmioty,

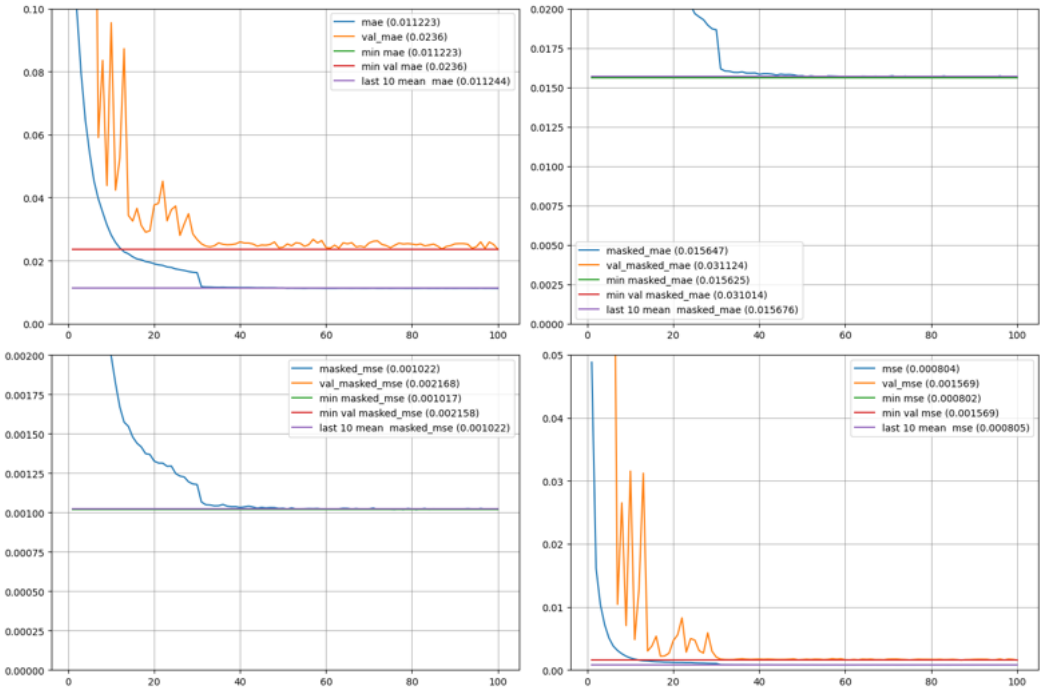

Fig. 4.3 Wykres metryk szkolenia sieci neuronoewj opartej an architekturze U-Net, której zadaniem jest stworzenie mapy głębokości na podstawie zjęcia sadzonek drzew w Lizymetrze. Oprócz standardowych miar, jak średni błąd kwadratowy (MSE) i (MAE), wyliczamy błędy "masked MSE" i "masked MAE", które wyliczają te same miary, ale biorąc pod uwagę tylko piksele wewnątrz lizymetru (wykorzystują semantyczną maskę do określenia istotnych pikseli)

rośliny. W wersji dostosowanej do skali metrycznej (metric fine-tuned), Depth Anything został przystosowany do pracy z danymi, gdzie precyzja w skali odległości jest kluczowa, co stanowiło istotny krok w jego rozwoju.

W artykule [5] opisano proces dostosowania tego modelu, w którym zastosowano dostrajanie na danych obrazowych o znanej geometrii i precyzyjnie oznaczonych mapach głębokości. Celem było zapewnienie, aby przewidywane mapy głębokości zachowywały realistyczne proporcje oraz odległości między obiektami w przestrzeni trójwymiarowej, co było kluczowe również w moich badaniach nad sadzonkami drzew.

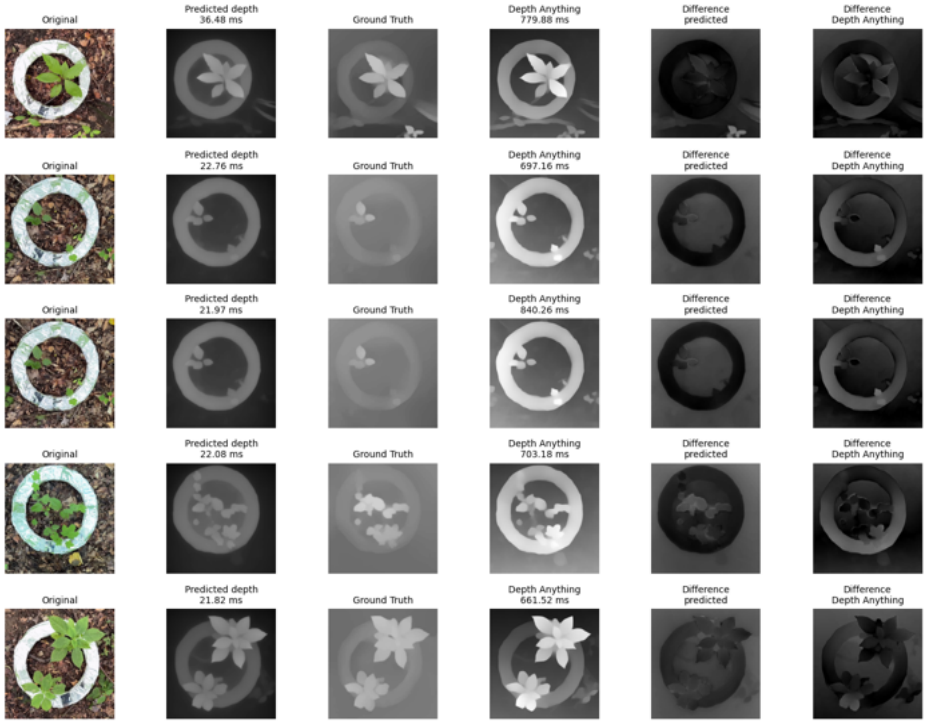

Fig. 4.4 Porównanie wyników estymacji głębokości uzyskanych za pomocą modelu Depth Anything dostrojonego do zdjęć sadzonek drzew w Lizymetrze. Pierwsza kolumna przedstawia oryginalny obraz. Druga kolumna prezentuje przewidywaną mapę głębokości wygenerowaną przez model Depth Anything po dostrojeniu. Trzecia kolumna pokazuje mapę głębokości uzyskaną za pomocą czujnika LiDAR iPhone'a, skalibrowaną w celu uchwycenia rzeczywistej głębokości w centymetrach, służącą jako prawdziwe odniesienie. Czwarta przedstawia wynik uzyskany przez podstawowy model Depth Anything. Piąta różnicę bezwzględną między mapą głębokości wygenerowaną przez model dostrojony a mapą prawdziwego odniesienia. Szósta przedstawia różnicę bezwzględną między mapą głębokości przed dostrojeniem a mapą odniesienia.

Korzystając z dostrojonej wersji modelu Depth Anything, przeprowadziłem dodatkowy fine-tuning w oparciu o dane syntetyczne, które samodzielnie wygenerowałem w środowisku Unity. Było to potrzebne, poniważ istniejące wersje tego modelu nie były wystarczające do tak specyficznego przykładu 4.6. Wielokrotnie obiekty znajdujące się blisko były oznaczane

jako obiekty znajdujące się daleko. W tym konkretnym zastosowaniu nawet model metryczny nie był wstanie poprawnie przewidzieć głębokości. Dane użyte do dalszego dostrojenia były podobne jak w przypadku szkolenia U-Net. Dane te obejmowały zarówno obrazy 2D, jak i precyzyjne mapy głębokości, które odwzorowywały sadzonki drzew w skali metrycznej. Dzięki temu model Depth Anything został dodatkowo zoptymalizowany do przewidywania map głębokości w specyficznych warunkach lizymetru, co z kolei pozwoliło na uzyskanie bardziej precyzyjnych wyników w moich eksperymentach.

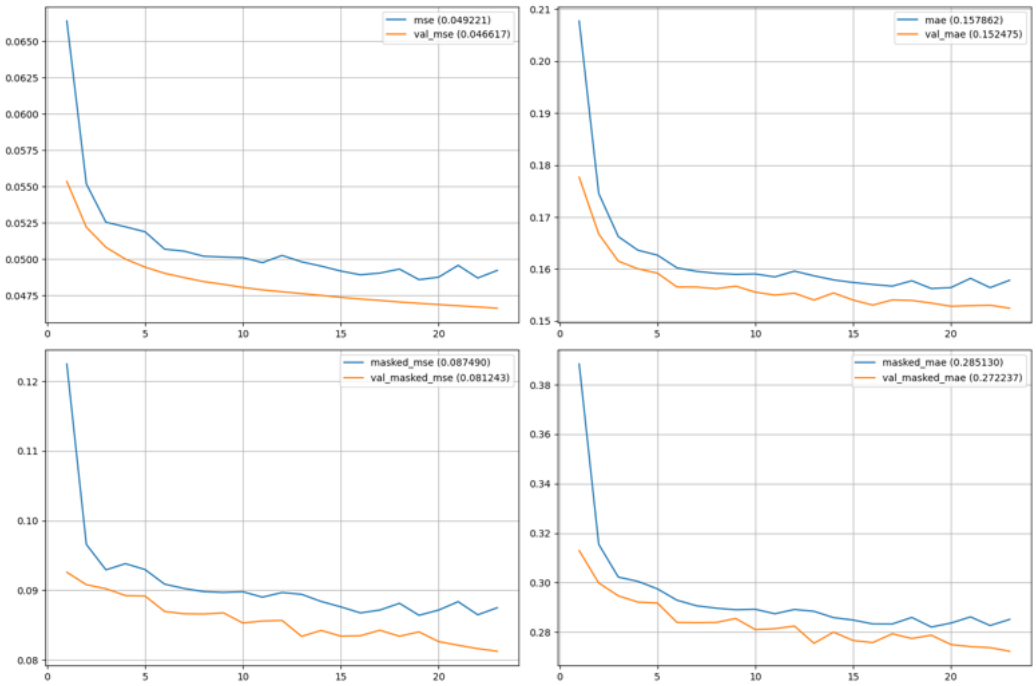

Fig. 4.5 Wykres metryk dostrajania modelu Depth Anything, której zadaniem jest stworzenie mapy głębokości na podstawie zdjęcia. Model zostaje dopasowany do zdjęć sadzonek drzew w lizymetrze. Oprócz standardowych miar, jak Średni Błąd Kwadratowy (MSE) i Średni Błąd Bezwzględny (MAE) wyliczane są błędy "masked MSE" i "masked MAE", które wyliczają te same miary, ale biorąc pod uwagę tylko piksele wewnątrz lizymetru

4.4 Analiza wyników

W tabeli 4.1 zaprezentowano wyniki modeli U-Net oraz Depth Anything, na zbiorze treningowym oraz walidacyjnym, w oparciu o metryki MSE, MAE oraz ich maskowane wersje.

Dostrojenie modelu Depth Anything przyniosło pozytywny skutek, i wyniki te są dużo lepsze. Jednak wynik modelu U-Net zdecydowanie osiągnął najlepsze wyniki. Jest to też

(a) Niedostrojony DA (b) Dostrojony DA (c) U-Net

Fig. 4.6 Porównanie danych prawdziwych z wynikami modeli regresji dla map głębokości lizymetru. Panel (a) przedstawia wyniki dla niedostrojonej wersji Depth Anything (DA), panel (b) dla dostrojonej wersji Depth Anything, a panel (c) dla modelu U-Net. Na niebiesko są oznaczone poszczególne wyniki, a na czerowno jest pokazana regresja liniowa. Zmiana w odległości tych punktów od lini regresji obrazuje poprawę dokładności wylicznania odległości do lizymetru. Również kierunek regresji liniowej jest istotny i przy idelanym wyniku powinien przebiegać pod kątem 45º (wartość rzeczywista odpowiada wartości przewidzianej)

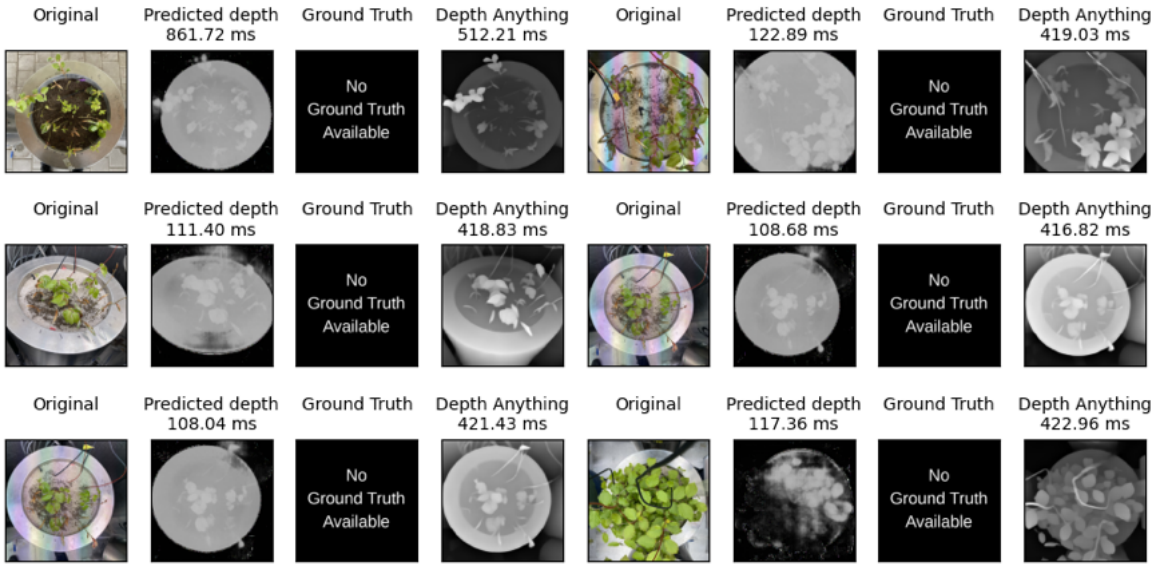

Fig. 4.7 Przykłady dla zdjęć rzeczywistych (model U-Net). Zdjęcia te nie posiadają informacji o poprawnym wyniku, ale były zrobione w trakcie trwania eksperymentu. Na tych zdęciach widać również limitacje tych modeli, takie jak wymaganie widoku z góry, oraz pewne ograniczenia co do ilości liści na zdjęciu

Tabela 4.1 Porównanie wyników modeli U-Net i Depth Anything (DA) trenowanych na syntetycznych oraz rzeczywistych danych

Model	Zbiór	MSE	MAE	Masked MSE/MAE
DA metryczny	Walidacyjny	0.1	0.288	0.234 / 0.629
DA metryczny	Testowy	0.097	0.281	0.226 / 0.616
DA metryczny dostrojony	Walidacyjny	0.0471	0.1533	0.0826 / 0.2748
DA metryczny dostrojony	Testowy	0.0437	0.1482	0.0761 / 0.2630
U-Net	Walidacyjny	0.0017	0.0246	0.0024 / 0.0324
U-Net	Testowy	0.0017	0.0239	0.0023 / 0.0317

najleżejszy model, który ma mniejsze wymagania srzętowe i przewiduje wynik zdecydowanie szybciej. Jednak i on nie jest perfekcyjny. Przechodząc do danych rzeczywistych Dostrojony model DA się okacuje dużo bardziej skuteczny. Zwłaszcza w sytuacjach, kiedy jset bardzo duże zagęszczenie obiektów w scenie (ponad 100 liści), to pozostałe modele nie wiedzą jak się zachować, a ten model wciąż podaję wyniki z podobną dokłądnością. Wykazuje się tym większą odpornością na zróżnicowane dane.

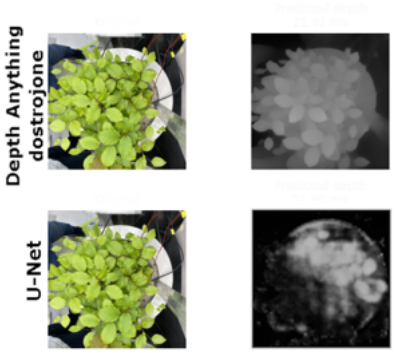

Fig. 4.8 Przykład, w którym U-Net nie dał spodziewanego rezultatu, za to dostrojona wersja Depth Anything szczegółowo odwzorowała głębokość liści.

Rysunki 4.2 oraz 4.9 prezentują porównanie wyników estymacji głębokości za pomocą modeli U-Net i Depth Anything na zestawach danych syntetycznych i rzeczywistych. Natępnie na rysunku 4.4 jest takie samo porównianie wyników dostrojonego modelu Depth Anything (w wersji metrycznej) z Depth Anything. Zauważalne są różnice w dokładności przewidywań, które są szczególnie istotne w kontekście analizy roślin w lizymetrze.

4.5 Wnioski

Przedstawiono rozszeżenie metody generowania danych treningowych dla sieci neuronowych, metody przedstawionej w poprzednim rozdziale. Powstały zbiór danych umożliwia

rekonstrukcje map głębokości, które mogą być następnie użyte do oszacowania kształtu i wymiarów wielu roślin na podstawie pojedynczego zdjęcia. Ulepszony zbiór zawiera dane o wyższej jakości, które również posiadają więcej opisów i informacji. Została przeprowadzona analiza wynkiów szkolenia różnych architektór sieci w oparciu o te dane, które wykazałą pozytywne wyniki.

Metoda ta otwiera możliwość wielu dalszych badań. W oparciu o powstały zbiór danych można opracować sieci przewidujące maski instancyjne, wyodrembniające poszczeólne liście. Następnie dla tych wyodrębnionych liści można zająć się problemem przewidywania rozmiaru liścia (podobnie jak w poprzednnim rozdziale, jednak teraz dla wielu liści na raz). Można też dążyć do odtworzenia całej struktury 3D wszystkich sadzonek widocznych na zdjęciu.

Kolejnymi aspektami przyszłych badań jest wykożystanie i sprawdzenie działania tej metody do innych zastosowań. Istnieje wiele problemów o podłożu biologicznym, ekologicznnym, czy nawet medycznym, które by mogły zostać rozwiązane w ten sposób, jako, że nie wymaga on wielu przykładów danych rzeczywistych do osiągnieća dobrych wyników.

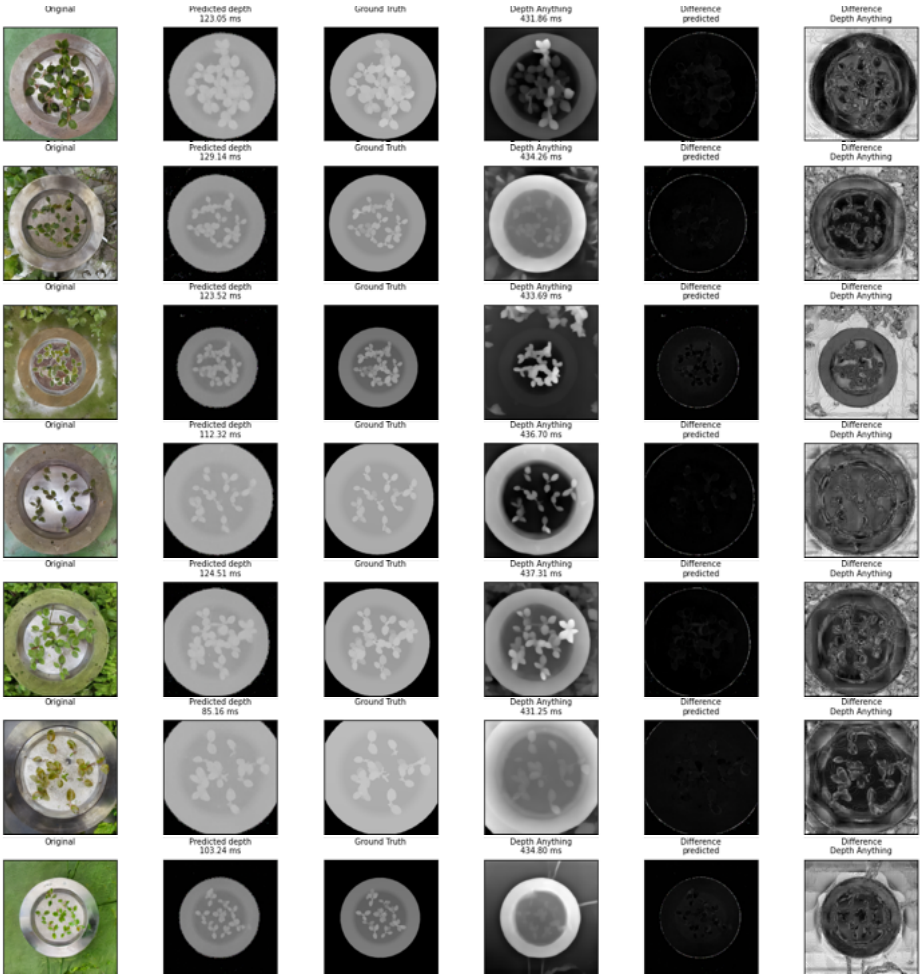

Fig. 4.9 Porównanie wyników estymacji głębokości za pomocą modeli U-Net i Depth Anything. Pierwsza kolumna przedstawia oryginalny obraz. Druga kolumna prezentuje przewidywaną mapę głębokości wygenerowaną przez model U-Net, dostosowany do zadań tłumaczenia obraz-obrazu. Trzecia pokazuje mapę głębokości uzyskaną za pomocą czujnika LiDAR iPhone'a, skalibrowaną w celu uchwycenia rzeczywistej głębokości w centymetrach. Czwarta przedstawia mapę głębokości wygenerowaną przez model Depth Anything, skonfigurowany do przewidywań głębokości w jednostkach metrycznych. Piąta i szósta ukazują różnicę bezwzględną między prawdziwą mapą głębokości a przewidywaniami modeli U-Net i Depth Anything, odpowiednio podkreślając obszary, gdzie przewidywania odbiegają od rzeczywistych pomiarów głębokości. Widać, że wewnątrz lizymetru różnice są mniejsze przy użyciu mojej metody w porównaniu do modelu Depth Anything.

Rozdział 5

Podsumowanie

W niniejszej rozprawie podjęto próbę rozwiązania wieloaspektowych wyzwań związanych z analizą danych obrazowych za pomocą metod widzenia maszynowego oraz głębokiego uczenia. Badania skupiały się na integracji sztucznych sieci neuronowych i modeli proceduralnych, aby umożliwić dokładną i efektywną analizę danych obrazowych, ze szczególnym uwzględnieniem, ale nie ograniczając się do, danych biologicznych. Szczególną rolę odegrało wykorzystanie powszechnie dostępnych urządzeń, takich jak smartfony, do zbierania danych, co ma istotne znaczenie w szerokim kontekście aplikacji, od rolnictwa po monitoring środowiskowy.

Hipoteza 1: Wykorzystanie sztucznych sieci neuronowych

Postawiona hipoteza zakładała, że sztuczne sieci neuronowe mogą być efektywnie wykorzystane do zbierania, ekstrakcji i analizy danych obrazowych. Przeprowadzone badania wykazały, że sieci neuronowe, zwłaszcza konwolucyjne sieci neuronowe (CNN), są wysoce efektywne w przetwarzaniu i interpretacji złożonych obrazów. Wykorzystując CNN, udało się osiągnąć znaczące postępy w takich zadaniach jak estymacja powierzchni obiektów oraz rekonstrukcja geometrii na podstawie pojedynczych fotografii. Modele te były trenowane na rozległych zbiorach danych, obejmujących zarówno obrazy rzeczywiste, jak i syntetyczne, co zapewniło ich odporność i dokładność w różnych warunkach.

Hipoteza 2: Modele proceduralne i syntetyczne zbiory danych

Druga hipoteza zakładała, że modele proceduralne generujące syntetyczne zbiory danych mogą rozwiązać problem małych zbiorów danych, co jest częstym wyzwaniem w analizie danych obrazowych. Małe zbiory danych rzeczywistych są w sposób szczególny problemem przy danych biologicznych. Wprowadzono różne syntetyczne zestawy danych, w tym zbiór danych LAESI, stworzony przy użyciu modeli proceduralnych i technik generatywnej

sztucznej inteligencji. Zbiory te dostarczyły wszechstronnego zasobu do trenowania modeli uczenia maszynowego, znacząco poprawiając ich wydajność w analizie danych obrazowych. Poprzez generowanie syntetycznych obrazów z precyzyjnymi adnotacjami, wykazałem, że dane syntetyczne mogą skutecznie uzupełniać rzeczywiste zbiory danych, redukując zależność od pracochłonnego ręcznego zbierania danych.

Hipoteza 3: Integracja danych syntetycznych i rzeczywistych

Zakładano, że integracja danych syntetycznych z rzeczywistymi poprawi odporność i ogólność modeli uczenia maszynowego. Przeprowadzone eksperymenty potwierdziły tę hipotezę, pokazując, że modele trenowane na kombinacji danych syntetycznych i rzeczywistych przewyższają te trenowane wyłącznie na danych rzeczywistych. Wykorzystanie danych syntetycznych nie tylko zwiększyło dokładność modeli, ale również dostarczyło skalowalnego i ekonomicznego rozwiązania dla generowania oznakowanych zbiorów danych. Ta integracja okazała się szczególnie korzystna w scenariuszach, gdzie rzeczywiste dane były rzadkie lub trudne do uzyskania, co dowodzi potencjału zbiorów danych syntetycznych w szerokim spektrum zastosowań, od biologii po inżynierię.

Badania skutecznie wykazały zastosowanie sztucznych sieci neuronowych oraz generowania danych syntetycznych w analizie danych obrazowych. Opracowane metody i zbiory danych mają istotne implikacje dla wielu dziedzin, w tym rolnictwa, monitoringu środowiskowego, inżynierii i innych. Przyszłe prace będą koncentrować się na rozszerzeniu zakresu generowania danych syntetycznych, uwzględniając bardziej zróżnicowane próbki oraz udoskonaleniu architektur sieci neuronowych w celu dalszego poprawienia ich dokładności i efektywności.

Podsumowując, niniejsza praca doktorska wniosła istotny wkład w dziedzinę analizy danych obrazowych, potwierdzając postawione hipotezy oraz dostarczając solidnych podstaw do dalszych badań. Wykorzystując moc uczenia maszynowego i danych syntetycznych, torujemy drogę do bardziej dostępnych, efektywnych i dokładnych metod analizy danych obrazowych.

Bibliografia

[1] Anagnostopoulou, D., Retsinas, G., Efthymiou, N., Filntisis, P., and Maragos, P. (2023). A realistic synthetic mushroom scenes dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 6282–6289.

[2] Anastacio, F., Sousa, M. C., Samavati, F., and Jorge, J. A. (2006). Modeling plant structures using concept sketches. In *Proceedings of the 4th International Symposium on Non-Photorealistic Animation and Rendering*, NPAR '06, page 105–113. Association for Computing Machinery.

[3] Aono, M. and Kunii, T. (1984). Botanical tree image generation. *IEEE Comput. Graph. Appl.*, 4(5):10–34.

[4] Argudo, O., Chica, A., and Andujar, C. (2016). Single-picture reconstruction and rendering of trees for plausible vegetation synthesis. *Comput. Graph.*, 57(C):55–67.

[5] Author, U. (2024). Depth anything: Scaling depth estimation for computer vision. *Journal of Artificial Intelligence Research*, pages 100–120.

[6] Behrendt, S., Colditz, C., Franzke, O., Kopf, J., and Deussen, O. (2005). Realistic real-time rendering of landscapes using billboard clouds. *Comp. Graph. Forum*, 24(3):507–516.

[7] Benes, B. and Millán, E. U. (2002). Virtual climbing plants competing for space. In *Proceedings of the Computer Animation*, CA '02, page 33, USA. IEEE Computer Society.

[8] Blozan, W. (2006). Tree measuring guidelines of the eastern native tree society.

[9] Bradley, D., Nowrouzezahrai, D., and Beardsley, P. (2013). Image-based reconstruction and synthesis of dense foliage. *ACM Trans. on Grap.*, 32(4):74:1–74:10.

[10] Brument, B., Bruneau, R., Quéau, Y., Mélou, J., Bernard, F., Lauze, F., Durou, J.-D., and Calvet, L. (2023). Rnb-neus: Reflectance and normal-based multi-view 3d reconstruction. *arXiv preprint arXiv:2312.01215*.

[11] Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, 8(6):679–698.

[12] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2016). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *CoRR*, abs/1606.00915.

[13] Chen, X., Neubert, B., Xu, Y.-Q., Deussen, O., and Kang, S. B. (2008). Sketch-based tree modeling using markov random field. *ACM Trans. on Grap.*, 27(5).

[14] Chiba, N., Ohshida, K., Muraoka, K., and Saito, N. (1996). Visual simulation of leaf arrangement and autumn colours. *The Journal of Visualization and Computer Animation*, 7(2):79–93.

[15] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE.

[16] Deussen, O., Colditz, C., Stamminger, M., and Drettakis, G. (2002). Interactive visualization of complex plant ecosystems. *VIS '02*, pages 219–226.

[17] Easlon, H. M. and Bloom, A. J. (2014). Easy leaf area: Automated digital image analysis for rapid and accurate measurement of leaf area. *Applications in Plant Sciences*, 2(7):1400033.

[18] Eigen, D. and Fergus, R. (2014). Depth map prediction from a single image using a multi-scale deep network. *Advances in Neural Information Processing Systems*, 27.

[19] Fadaeddini, A., Majidi, B., and Eshghi, M. (2018). A case study of generative adversarial networks for procedural synthesis of original textures in video games. In *2018 2nd National and 1st International Digital Games Research Conference: Trends, Technologies, and Applications (DGRC)*, pages 118–122. IEEE.

[20] Fan, L., Chen, K., Krishnan, D., Katabi, D., Isola, P., and Tian, Y. (2023). Scaling laws of synthetic images for model training ... for now.

[21] Fei, J., Chen, L., Xu, Y., and He, Y. (2021a). Enlisting 3d crop models and gans for more data-efficient agricultural research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 1234–1242.

[22] Fei, Z., Olenskyj, A., Bailey, B. N., and Earles, M. (2021b). Enlisting 3d crop models and gans for more data efficient and generalizable fruit detection. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 1269–1277, Los Alamitos, CA, USA. IEEE Computer Society.

[23] Goldman, R., Schaefer, S., and Ju, T. (2004). Turtle geometry in computer graphics and computer-aided design. *Computer-Aided Design*, 36:1471–1482.

[24] Goss, M. J. and Ehlers, W. (2009). The role of lysimeters in the development of our understanding of soil water and nutrient dynamics in ecosystems. *Soil Use and Management*, 25(3):213–223.

[25] Greene, N. (1989). Voxel space automata: Modeling with stochastic growth processes in voxel space. *SIGGRAPH Comp. Graph.*, 23(3):175–184.

[26] Guo, J., Jiang, H., Benes, B., Deussen, O., Zhang, X., Lischinski, D., and Huang, H. (2020). Inverse procedural modeling of branching structures by inferring l-systems. *ACM Trans. on Grap.*, 39(5).

[27] Guzdial, M., Long, D., Cassion, C., and Das, A. (2020). Visual procedural content generation with an artificial abstract artist. *Proceedings of ICCC Computational Creativity and Games Workshop*.

[28] Ha, D. and Eck, D. (2017). A neural representation of sketch drawings. *arXiv preprint arXiv:1704.03477*.

[29] Habel, R., Kusternig, A., and Wimmer, M. (2009). Physically guided animation of trees. *Comp. Graph. Forum*, 28(2):523–532.

[30] Hädrich, T., Benes, B., Deussen, O., and Pirk, S. (2017). Interactive modeling and authoring of climbing plants. *Comput. Graph. Forum*, 36(2):49–61.

[31] Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA.

[32] He, R., Sun, S., Yu, X., Xue, C., Zhang, W., Torr, P., Bai, S., and Qi, X. (2023). Is synthetic data from generative models ready for image recognition? In *International Conference on Learning Representations*.

[33] Honda, H. (1971). Description of the form of trees by the parameters of the tree-like body: Effects of the branching angle and the branch length on the shape of the tree-like body. *Journal of Theoretical Biology*, 31(2):331 – 338.

[34] Hong, S. M., Simpson, B., and VG Baranoski, G. (2005). Interactive venation-based leaf shape modeling. *Computer Animation and Virtual Worlds*, 16(3-4):415–427.

[35] Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q. V., and Adam, H. (2019). Searching for mobilenetv3. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[36] Hu, E. et al. (2021). Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

[37] Ijiri, T., Owada, S., and Igarashi, T. (2006). Seamless integration of initial sketching and subsequent detail editing in flower modeling. *Comp. Graph. Forum*, 25(3):617–624.

[38] Karis, B. (2013). Real shading in unreal engine 4. Technical report, Epic Games.

[39] Kawaguchi, Y. (1982). A morphological study of the form of nature. *SIGGRAPH Comp. Graph.*, 16(3):223–232.

[40] Klein, J., Waller, R. E., Pirk, S., Pałubicki, W., Tester, M., and Michels, D. L. (2023). Synthetic data at scale: A paradigm to efficiently leverage machine learning in agriculture. *Available at SSRN 4314564*.

[41] Kratt, J., Spicker, M., Guayaquil, A., Fišer, M., Pirk, S., Deussen, O., Hart, J. C., and Benes, B. (2015). Woodification: User-controlled cambial growth modeling. *CGF*, 34(2):361–372.

[42] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 25, pages 1097–1105. Curran Associates, Inc.

[43] Li, C., Deussen, O., Song, Y.-Z., Willis, P., and Hall, P. (2011). Modeling and generating moving trees from video. *ACM Trans. on Grap.*, 30(6):127:1–127:12.

[44] Li, Y., Fan, X., Mitra, N. J., Chamovitz, D., Cohen-Or, D., and Chen, B. (2013). Analyzing growing plants from 4d point cloud data. *ACM Trans. on Grap.*, 32(6).

[45] Lintermann, B. and Deussen, O. (1999). Interactive modeling of plants. *IEEE Comput. Graph. Appl.*, 19(1):56–65.

[46] Livny, Y., Pirk, S., Cheng, Z., Yan, F., Deussen, O., Cohen-Or, D., and Chen, B. (2011a). Texture-lobes for tree modelling. *ACM Trans. on Grap.*, 30(4):53:1–53:10.

[47] Livny, Y., Yan, F., Olson, M., Chen, B., Zhang, H., El-Sana, J., and Funkhouser, T. (2011b). Automatic reconstruction of tree skeletal structures from point clouds. *ACM Transactions on Graphics*, 30(6):1510–1518.

[48] Longay, S., Runions, A., Boudon, F., and Prusinkiewicz, P. (2012). Treesketch: interactive procedural modeling of trees on a tablet. In *Proc. of the Intl. Symp. on SBIM*, pages 107–120.

[49] McInnes, L., Healy, J., and Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

[50] Michels, D. L., Mueller, J. P. T., and Sobottka, G. A. (2015). A physically based approach to the accurate simulation of stiff fibers and stiff fiber meshes. *Comput. Graph.*, 53:136–146.

[51] Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., and Ng, R. (2020). Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, pages 405–421.

[52] Měch, R. and Prusinkiewicz, P. (1996). Visual models of plants interacting with their environment. In *Proc. of SIGGRAPH*, pages 397–410. ACM.

[53] Neubert, B., Franken, T., and Deussen, O. (2007a). Approximate image-based tree-modeling using particle flows. *ACM Transactions on Graphics*, 26(3):88.

[54] Neubert, B., Franken, T., and Deussen, O. (2007b). Approximate image-based tree-modeling using particle flows. *ACM Trans. on Grap.*, 26(3).

[55] Neubert, B., Pirk, S., Deussen, O., and Dachsbacher, C. (2011). Improved model- and view-dependent pruning of large botanical scenes. *Comp. Graph. Forum*, 30(6):1708–1718.

[56] Nguyen, Q., Vu, T., Tran, A., and Nguyen, K. (2023). Dataset diffusion: Diffusion-based synthetic data generation for pixel-level semantic segmentation. *Advances in Neural Information Processing Systems (NeurIPS)*.

[57] Okabe, M., Owada, S., and Igarashi, T. (2007). Interactive design of botanical trees using freehand sketches and example-based editing. In *ACM SIGGRAPH Courses*. ACM.

[58] Oppenheimer, P. E. (1986). Real time design and animation of fractal plants and trees. *Proc. of SIGGRAPH*, 20(4):55–64.

[59] Palubicki, W., Horel, K., Longay, S., Runions, A., Lane, B., Měch, R., and Prusinkiewicz, P. (2009). Self-organizing tree models for image synthesis. *ACM Trans. on Grap.*, 28(3):58:1–58:10.

[60] Pirk, S., Benes, B., Ijiri, T., Li, Y., Deussen, O., Chen, B., and Měch, R. (2016). Modeling plant life in computer graphics. In *ACM SIGGRAPH 2016 Courses*.

[61] Pirk, S., Jarząbek, M., Hädrich, T., Michels, D. L., and Palubicki, W. (2017). Interactive wood combustion for botanical tree models. *ACM Trans. on Grap.*, 36(6):197:1–197:12.

[62] Pirk, S., Niese, T., Deussen, O., and Neubert, B. (2012a). Capturing and animating the morphogenesis of polygonal tree models. *ACM Trans. on Grap.*, 31(6):169:1–169:10.

[63] Pirk, S., Niese, T., Hädrich, T., Benes, B., and Deussen, O. (2014). Windy trees: Computing stress response for developmental tree models. *ACM Trans. on Grap.*, 33(6):204:1–204:11.

[64] Pirk, S., Stava, O., Kratt, J., Said, M. A. M., Neubert, B., Měch, R., Benes, B., and Deussen, O. (2012b). Plastic trees: interactive self-adapting botanical tree models. *ACM Trans. on Grap.*, 31(4):50:1–50:10.

[65] Prusinkiewicz, P. and Lindenmayer, A. (1990). *The Algorithmic Beauty of Plants*. Springer-Verlag New York, Inc.

[66] Quan, L., Tan, P., Zeng, G., Yuan, L., Wang, J., and Kang, S. B. (2006). Image-based plant modeling. *ACM Trans. on Grap.*, 25(3):599–604.

[67] Quigley, E., Yu, Y., Huang, J., Lin, W., and Fedkiw, R. (2018). Real-time interactive tree animation. *IEEE Trans. on Vis. and Comp. Graphics*, 24(5):1717–1727.

[68] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*.

[69] Rahimikhoob, H., Delshad, M., and Habibi, R. (2023). Leaf area estimation in lettuce: Comparison of artificial intelligence-based methods with image analysis technique. *Measurement*, 222:113636.

[70] Raistrick, A., Lipson, L., Ma, Z., Mei, L., Wang, M., Zuo, Y., Kayan, K., Wen, H., Han, B., Wang, Y., Newell, A., Law, H., Goyal, A., Yang, K., and Deng, J. (2023). Infinite photorealistic worlds using procedural generation. *arXiv preprint arXiv:2306.09310*. CVPR 2023.

[71] Reche-Martinez, A., Martin, I., and Drettakis, G. (2004). Volumetric reconstruction and interactive rendering of trees from photographs. *ACM Trans. on Grap.*, 23(3):720–727.

[72] Reeves, W. T. and Blau, R. (1985). Approximate and probabilistic algorithms for shading and rendering structured particle systems. *SIGGRAPH Comput. Graph.*, 19(3):313–322.

[73] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

[74] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *CVPR*.

[75] Runions, A., Fuhrer, M., Lane, B., Federl, P., Rolland-Lagan, A.-G., and Prusinkiewicz, P. (2005). Modeling and visualization of leaf venation patterns. In *ACM SIGGRAPH 2005 Papers*, pages 702–711.

[76] Runions, A., Lane, B., and Prusinkiewicz, P. (2007). Modeling trees with a space colonization algorithm. *EG Nat. Phenom.*, pages 63–70.

[77] Runions, A., Tsiantis, M., and Prusinkiewicz, P. (2017). A common developmental program can produce diverse leaf shapes. *New Phytologist*, 216(2):401–418. Special Issue: Plant developmental evolution.

[78] Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C. W., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al. (2022). Laion-5b: An open large-scale dataset for training next generation image-text models. In *Advances in Neural Information Processing Systems*.

[79] Shao, H., Kugelstadt, T., Hädrich, T., Pałubicki, W., Bender, J., Pirk, S., and Michels, D. L. (2021). Accurately solving physical systems with graph learning.

[80] Shipard, J., Wiliem, A., Thanh, K. N., Xiang, W., and Fookes, C. (2023). Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion.

[81] Smith, A. R. (1984). Plants, fractals, and formal languages. In *Proc. of SIGGRAPH*, pages 1–10. ACM Press.

[82] Song, J., Lu, Y., Chen, X., and Wang, H. (2022). A comprehensive survey on generative diffusion models for structured data. *arXiv preprint arXiv:2206.04139*.

[83] Stava, O., Pirk, S., Kratt, J., Chen, B., Měch, R., Deussen, O., and Benes, B. (2014). Inverse procedural modelling of trees. *Comp. Graph. Forum*, 33(6):118–131.

[84] Sun, T., Segu, M., Postels, J., Wang, Y., Van Gool, L., Schiele, B., Tombari, F., and Yu, F. (2022). Shift: A synthetic driving dataset for continuous multi-task domain adaptation. In *CVPR*.

[85] Tan, P., Fang, T., Xiao, J., Zhao, P., and Quan, L. (2008). Single image tree modeling. *ACM Trans. on Grap.*, 27(5):108:1–108:7.

[86] Tan, P., Zeng, G., Wang, J., Kang, S. B., and Quan, L. (2007). Image-based tree modeling. *ACM Trans. on Grap.*, 26(3).

[87] Ubbens, J., Cieslak, M., Prusinkiewicz, P., and Stavness, I. (2018). The use of plant models in deep learning: an application to leaf counting in rosette plants. *Plant methods*, 14:1–10.

[88] Wang, B., Zhao, Y., and Barbič, J. (2017). Botanical materials based on biomechanics. *ACM Trans. on Grap.*, 36(4):135:1–135:13.

[89] Wang, I. R., Wan, J. W., and Baranoski, G. V. (2004). Physically-based simulation of plant leaf growth. *Computer Animation and Virtual Worlds*, 15(3-4):237–244.

[90] Ward, D., Moghadam, P., and Hudson, N. (2018). Deep leaf segmentation using synthetic data. In *British Machine Vision Conference (BMVC) 2018, CVPPP Workshop*. arXiv:1807.10931 [cs.CV].

[91] Watson, D. J. (1947). Comparative Physiological Studies on the Growth of Field Crops: I. Variation in Net Assimilation Rate and Leaf Area between Species and Varieties, and within and between Years. *Annals of Botany*, 11(1):41–76.

[92] Wither, J., Boudon, F., Cani, M.-P., and Godin, C. (2009). Structure from silhouettes: a new paradigm for fast sketch-based design of trees. *Comp. Graph. Forum*, 28(2):541–550.

[93] Xu, H., Gossett, N., and Chen, B. (2007). Knowledge and heuristic-based modeling of laser-scanned trees. *ACM Trans. on Grap.*, 26(4):Article 19, 13 pages.

[94] Yan, Q., Zheng, J., Reding, S., Li, S., and Doytchinov, I. (2021). Crossloc: Scalable aerial localization assisted by multimodal synthetic data. *arXiv preprint arXiv:2112.09081*.

[95] Yu, Z., Zhu, C., Culatana, S., Krishnamoorthi, R., Xiao, F., and Lee, Y. J. (2023). Diversify, don't fine-tune: Scaling up visual recognition training with synthetic images.

[96] Zhang, L. and Agrawala, M. (2023). Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:ToBeUpdated*.

[97] Zhang, L., Rao, A., and Agrawala, M. (2023). Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*.

[98] Zhang, Y., Ling, H., Gao, K., Yin, J., Lafleche, J., Barriuso, A., Torralba, A., and Fidler, S. (2021). Datasetgan: Efficient labeled data factory with minimal human effort. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[99] Zhao, Y. and Barbič, J. (2013). Interactive authoring of simulation-ready plants. *ACM Trans. on Grap.*, 32(4):84:1–84:12.