SYDE 462 - Canopii

Team 19: Derin Denizkusu, Ethan Liang, Kevin Luong, Jessie Won

Abstract – Kitchener has created a Sustainable Urban Forest Strategy aimed towards improving their urban forest in order to achieve a more sustainable city [1]. This plan lacks a maintained tree inventory which inhibits effective long-term planning, especially in private areas. The project objective is to create a product used by residents in order to expand the existing tree inventory through facilitating data collection on private lands. The proposed solution is an application that extracts tree species and location from images and shares it with the city. The application allows users to take photos, determine the species, and add location data for a tree. The server then ingests this data, processes images using a convolutional neural network and support vector machine-based image recognition algorithm, and stores the result in a database. Testing has been performed on manual and automatic species identification flows using leaf images. Next steps include testing on real trees and leaves, integrating with Kitchener's Street Tree Inventory, and gathering more field data to improve the image recognition algorithm.

Keywords – Tree mapping, urban forestry, sustainability

I. INTRODUCTION

Kitchener has created a Sustainable Urban Forest Strategy aimed towards "planning, engaging, maintaining, protecting and planting Kitchener's urban forest" to combat climate change and improve city sustainability [1]. A gap in the existing program is the lack of tree data on private lands, which composes 56% of their urban forest [2]. The city has "no information about privately owned trees", which prevents them from using environmental modelling software vital to effective long-term urban forest planning [2]. The current process for gathering data for the tree inventory is individual manual tree tagging with survey wheels and dichotomous keys for tree identification [3]. Existing inventory solutions, such as ArcGIS and TreesCount! do not automate data collection [3, 4]. Existing automated data collection products, such as PlantSnap and LeafSnap, keep their data private [5, 6]. LiDAR is another method of data collection, but no public product exists and it cannot identify species [7,8].

Canopii's goal is to expand the existing tree inventory by focusing on private trees and facilitating data collection by enlisting home and property owners. This will, at a minimum, expand their inventory with information on private trees, which requires species and location data, to help cities plan their urban forest. Further development includes additional data extraction such as diameter breast height (DBH) and crown health, which would better enable the use of environmental modelling software such as i-Tree Eco [9]. Finally, Canopii aims to engage local residents, proven to instill a deeper understanding of sustainability issues, reduce long-term costs, and increase community capacity to mitigate climate change [10].

II. PROJECT SCOPE AND OBJECTIVES

Canopii is motivated by climate change; Earth's global temperature is on track for a 2°C rise, resulting in irreversible feedback loops causing widespread droughts, famines, and extreme weather events [11]. Healthy urban forests are a United Nations (UN) recommended solution to mitigate these hazards, though long-term planning is required to effectively build them [12]. Without a comprehensive and up-to-date inventory, cities cannot develop accurate plans. For example, cities need to track and maintain forest diversity in order to minimize the risk of invasive species. They also need to determine the rates of growth and decay to optimize where to plant future trees, what species to plant, and when to plant them.

The objectives of this project can be summarized in the following situation impact statement: *Design a product that is to be used by residents in Kitchener with access to privately owned trees in order to tag them and improve the city's tree inventory with data on tree species and location.* The success of this project will foster social engagement around sustainability and enable long-term planning to improve urban canopy in Kitchener [13, 14].

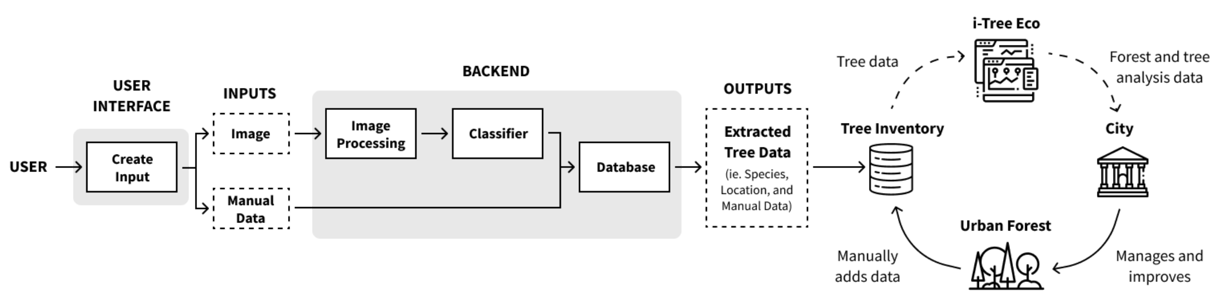


Figure 1: Systems diagram of project scope

The system diagram above shows the project scope and how it will be integrated with the current situation of concern. The section on the right shows the current system, where Kitchener manually adds data from their urban forest into their tree inventory. They do not have enough data to use their desired modelling software, i-Tree Eco, and cannot perform extensive analyses [15]. The proposed solution, Canopii, extracts tree data (output) from the user's entry (input). The main system components are the user interface (UI), where users can input data, and the back-end, which extracts additional information using image recognition and stores it into a database to be integrated to Kitchener.

III. SUMMARY OF ENGINEERING ANALYSIS AND DESIGN METHODS

USER INTERFACE

An iterative design methodology was used throughout the project, with a focus on the Prototype, Build, and Testing stages in SYDE 462 [16]. Since the user flow and features were already defined in SYDE 461, the approach for UI design was focused on rapid prototyping and testing to gain as much user feedback as possible. This feedback was then integrated into the designs using G Suite (Docs, Sheets, Slides) for documentation, Figma for design, InVision for prototyping, and React-Native for development, which will be further explained below.

Table 1: Summary of Engineering and Design Methods for UI Design

Stage	Methods
Research	Background research, expert interviews, competitive analysis
Ideation	Brainstorming, personas, concept sketches, wireframes
Prototype	Interaction and visual design, mockups
Build	Interactive prototypes, front-end and back-end development
Testing	Usability testing, A/B testing, ranking and weighting

SOFTWARE

Software analysis techniques from SYDE 322 Software Design were performed to design the system. Firstly, requirements elicitation was performed to determine the actors, scenarios, use cases and requirements of the system. Then, the analysis object model was used to translate the use cases and requirements into entities, control and boundary objects. Entities represent “domain specific objects” such as the user’s tree, while control and boundaries are “an object that executes user commands” and “an interface between an actor and a control” respectively [17]. This results of the software analysis object model is shown in Table 2 below.

Table 2: Software Analysis Object Model

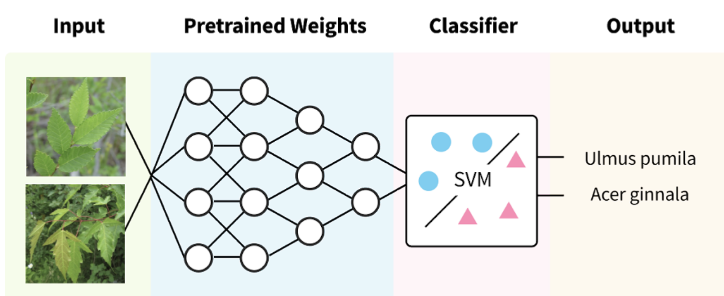
Entities	Boundaries	Controls
User User’s Tree (location, species, tree type) Database Entry Server	Application UI Take picture button Submit data button Manual data entry form	Get phone location Take picture Query data in database Process image Send image to server

Each object was then divided such that it is controlled by either the front-end or back-end. The front-end was designed to encompass all boundaries and controls involving data collection, since it requires user interaction. The back-end involves handling the database and server entities, as well as saving, querying and processing image data.

IMAGE RECOGNITION

The image recognition component of the solutions was created using practices developed in SYDE 372 and SYDE 522. A deep convolutional neural network (CNN) was chosen to extract image features from leaf images as they are best equipped for problems related to image recognition [18]. InceptionV3 was the chosen CNN based on performance and the network's public availability [19]. After experimentation with various different classifiers, a support vector machine (SVM) classifier was selected. The data was then split 80/20 into a training set and a testing set, as recommended by most image recognition problems regarding plant species classification [6, 20, 21, 22].

The original image recognition component was developed using leaf images scraped from Google Images for the 60 species provided by the Government of Ontario [23]. The classifier provides either a single suggested species or a list of suggested species, such as the top five or top 10 recommendations. Its accuracy was measured by the percentage that the correct species will be in the top N estimates. The accuracies from this solution are shown in Figure 2, which was not sufficiently accurate for the desired 80% top five accuracy defined in SYDE 461.



Accuracy	Original	Final
Top 1	43%	55%
Top 5	76%	91%
Top 10	87%	96%

Figure 2: Image recognition system with the accuracies of two prototypes compared

Upon further investigation, the list of tree species was too broad according to David Schmitt, Kitchener's Head of Urban Forestry. Instead, he provided a list of 34 species, described as the "best information" the city had on their urban forest composition [24]. This list covered all street and park trees as long as the species covered at least 0.4% of the city's canopy [24]. Since conifers are difficult to visually distinguish, the team only collected images for the 27 deciduous trees in this list. This time, "research-grade" images from iNaturalist, a crowd-sourced wildlife species observation platform, were used [25]. Images were carefully selected and edited to ensure all images had the same orientation and aspect ratio. The SVM classifier was then trained and optimized, using the recommended method of grid search on the parameters of C and gamma [26]. Accuracy results are summarized in Figure 2 above.

MANUAL SPECIES IDENTIFICATION KEY

Another engineering focus was designing and building a manual flow for species identification using custom dichotomous keys. The goal of this feature was to ensure accurate species data can be collected by untrained users with little taxonomic knowledge and would be used as additional verification to image recognition. Conifers could also be tagged using the manual flow since they could not be identified with Canopii's image recognition solution. A trait identification key was created for all 34 species using tree characteristics, or traits, which are the industry standard for species identification [27]. Books such as Trees in Canada and Trees of Ontario, which have dichotomous keys to identify various subsets of trees based on branching patterns and leaf features, were used as reference [27, 28].

By leveraging existing dichotomous keys, the data for 28 traits was collected for Canopii's tree list. Traits and trees were hierarchically organized to find various combinations which would narrow down the tree list in the shortest amount of steps. Accuracy for users selecting the right set of traits and the correct tree was determined for each combination, and the combination with the highest accuracy and least steps was selected. This combination was composed of 20 traits, five of which are various leaf shapes. Figure 3 shows the final identification key where each rectangle is a trait and the dotted rectangles represent the final groups of trees. The highlighted path shows the user having a deciduous tree with compound leaves and opposite branching. This narrows the species down to Acer negundo, Fraxinus pennsylvanica, and Fraxinus americana.

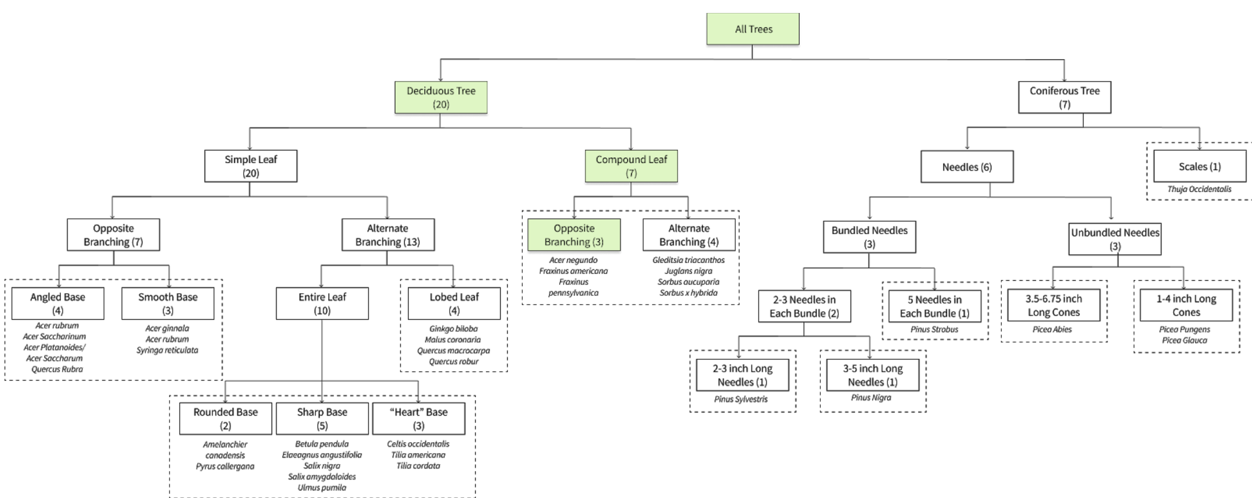


Figure 3: Manual key species breakdown

IV. SUMMARY OF SOCIAL, ECONOMIC, AND ENVIRONMENTAL IMPACTS

The main impact of this project will be to the local environment by providing a higher quality urban forest management plan driven by a comprehensive tree inventory. As previously mentioned, this inventory is fundamental to effective urban forest management [29]. The environmental damage in urban areas from climate change induced flooding and heat waves will cost an estimated $30 billion per year according to the Insurance Bureau of Canada [30, 31]. However, it would only cost around $5 billion per year to prevent that damage with investments into sustainable initiatives [31]. Urban forests are an initiative that mitigates these hazards, and Canopii automates tree inventory data collection for the urban forest management process, further improving the net impact of climate change prevention investment [12].

Climate change will also damage food supply chains [11]. The UN has recommended a food system that partially relies on urban forests to mitigate this issue [12]. A food plan reliant on urban forests requires knowledge of what trees are providing food in what locations, which can be tracked through the species and location information collected by Canopii, giving cities the tools to adapt to a new food supply infrastructure [32].

When municipalities invest in their urban forest, they also invest in sustainable job opportunities and increase engagement by locals. The UN has highlighted the job growth potential for urban forest initiatives, noting that the new job growth could provide career prospects to historically underemployed populations such as young people and women [33]. However, municipalities require tangible, measurable benefits in order to commit resources to these initiatives. Canopii collects the data that makes it possible for tools, such as i-Tree Eco, to calculate these benefits through metrics including carbon sequestration, hydrology improvements, building energy savings, pollution removal and human health impacts [9]. Canopii also works to increase community engagement in climate positive actions. By having users contribute to fighting climate change, citizens become engaged and aware of the cause. Social engagement has proven benefits including higher awareness surrounding sustainability issues and an increased community capacity to fight climate change [34].

A potential negative impact of Canopii is privacy violation if not correctly managed. An address is recorded in addition to images taken by users. To prevent violations, users are urged not to take pictures of sensitive data and explicitly notified of which data will be used publicly. Furthermore, since this data is not tied to a name, there is nothing to tie sensitive information to individuals. The automation of tree inventory data collection could also result in a number of jobs lost, but most of this work is currently being conducted by volunteers or city employees who can be allocated to other urban forest management tasks.

V. DESIGNED SOLUTION

MOBILE APPLICATION

Canopii is a mobile application for untrained users to tag trees. This was built using React-Native to increase the portability across operating systems and to allow for reusable UI components [35]. Redux was used for a persistent store of state to be kept across components, which was critical when populating data [36]. The application improves upon the current process by providing intuitive manual species identification for all trees, and automatic identification for deciduous trees. The manual flow is digitized and involves the traversal of an identification key. Illustrations were created using Figma for each trait, and users are presented the options and a description.

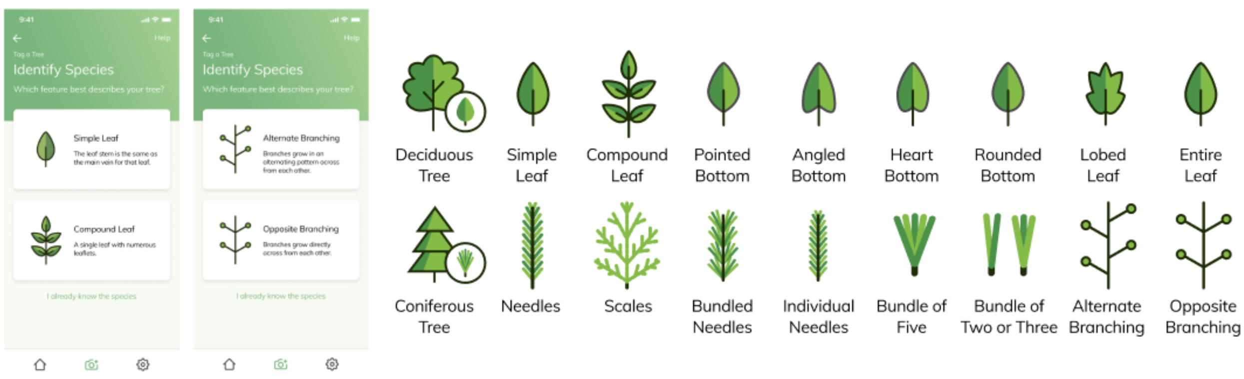


Figure 4: Manual tree identification trait illustrations

The automatic flow requires a photo capture which is sent to the server for processing. Afterwards, a subset of top five trees is returned so users can select the correct species. A species search option is also provided, so users can tag trees not in the preset list.

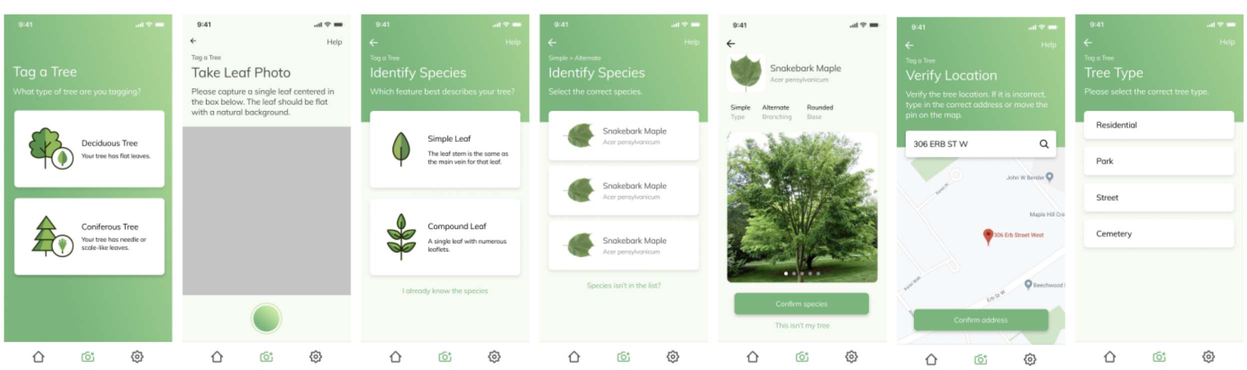


Figure 5: Tree tagging user flow

These approaches are better than the existing species solution as dichotomous keys are frustrating and difficult to use by non-experts [37]. The application also improves upon location, automatically obtained through the phone's GPS, and tree type, where users from a predefined list. This data is stored in a database for Kitchener to access.

BACK-END

To minimize integration overhead between the back-end and image recognition components, Django was chosen for the back-end server [38]. Several API endpoints were implemented for data entry, data query, and image processing which are activated on a POST request from the application. An AWS S3 bucket is used to store the images and is connected to a PostgreSQL database with three tables. Communication between the back-end and database are sent through a socket. The infrastructure was built on Docker, an off-the-shelf containerization engine, for easy portability, and is hosted on AWS.

Table 3: Database Tables and Associated Information

Table	**Stored Information**
TreeDB	Location, species, device of all trees tagged
SpeciesDB	Central source of truth for all tree data
ImageDB	Matches uploaded images with their storage

The back-end also hosts the image recognition algorithm which was developed in Keras with Google's InceptionV3. A CNN is used to extract image features from the leaf images, which are used by the SVM classifier to distinguish between species. The final SVM parameters were 30 for C and 0.0046 for gamma, which resulted in a top five accuracy of 90.5%.

VI. SUMMARY OF DESIGN EVALUATION

USER INTERFACE

The UI design evaluation was composed of multiple stages per an iterative design process to get as much user feedback as possible. The main validation method was in-person usability testing using the protocol [39]:

1. Identify goal and prepare respective prototypes, tasks, and questionnaires
2. Introduce yourself, the project, purpose, allow user to ask questions, build a rapport
3. Read the user the scenario and their task
4. Share prototype with user and ask them to think aloud as they walk through it

5. Record the user's actions, comments, questions, and body language
6. Share a post-test survey with reflective questions (how easy was, did you notice, etc)

In 461, the team tested the automated flow with an InVision prototype and received estimates on the amount of time users were willing to commit. For the next usability testing round, the goal was to test the manual identification key for deciduous trees. Testing was conducted with a slide deck of leaf and branch images, and was administered to testers using a document highlighting the various options and traits. Five users were tested on taking the correct path through the "Manual Key Path Test", and were tested on identifying the correct species out of a group results in the "Manual Key Group Test". These paths and groups can be referenced in Figure 3, and the results are shown in Table 4.

Table 4: Manual Dichotomous Key Testing Results

Manual Key Path Test

	Correct	Incorrect	Accuracy	Avg
Test 1	1	3	25%	50%
Test 2	1	3	25%	
Test 3	2	2	50%	
Test 4	4	0	100%	
Test 5	2	2	50%	

Manual Key Group Test

Correct	Incorrect	Accuracy	Avg
8	0	100%	80%
6	2	75%	
5	3	62.5%	
7	1	87.5%	
6	2	75%	

The results of the testing demonstrated that the ambiguity of certain early prompts in the path were able to completely mislead testers to an incorrect answer. While users were generally able to discern the correct species from a list, the path to arriving at the correct list is crucial. As the manual path was intended to be additional verification for image recognition, its required accuracy could not be lower. Working off of this result, the copy was changed and path prompts were coupled with accompanying visuals for improved clarity.

In the next round of testing, six users were given the application and asked to tag trees both manually and automatically. Quantitative data was collected in the form of time taken on each screen and the accuracy of identified species, as reflected by Table 5. Qualitative data was synthesized from user mistakes and comments. Quantitative data proved to be inconclusive as different users took vastly different amounts of time for different screens. Accuracy was also inconclusive as mistakes were made in different parts of the flow. This is likely due to the wide variety of technical prowess and age of the testers which ranged from 22-49. Comments from users indicated that the coniferous flow was especially confusing.

Users identified misleading icons, ambiguous instructions and lack of reference images as common pain points.

Table 5: Time Taken for Screens in Manual Flows

Deciduous	Simple v. Compound	Alternate v. Opposite	Select Species
Time taken (s)	5, 10, 23, 5, 5, 11	5, 10, 30, 5, 4, 20	30, 15, 8, 3, 2, 22

Coniferous	Bundled v. Individual	Scales v. Needles	Select Species
Time taken (s)	5, 20, 11, 18, 2, 35	3, 3, 20, 2, 31, 11	8, 20, 16, 2, 15, 13

This verified that the coniferous flow did not meet accuracy requirements. Hence, the coniferous flow was redesigned alongside small changes to the rest of the application. Next steps for UI evaluation are tagging live trees with leaves, which was inaccessible during the course of this project due to the winter season. Telemetry should also be implemented to send data to the server and enable users to report issues directly.

SOFTWARE

The software components were verified using unit testing and integration testing utilizing Jest, a front-end unit testing framework, and Django's test runner on the back-end. The back-end function of image upload and front-end components also have unit tests implemented to ensure there is no regression during development, along with a test plan to have 80% coverage for the whole system, meeting industry standards [40]. The image recognition component is divided into functions for loading the pre-trained neural network, loading the trained SVM classifier, extracting image features using the neural network, and classifying a leaf image into a list of top N species, all of which have unit tests. Finally, integration tests were written to verify that the link between application and server for image upload, species recognition, and data storage work as expected.

IMAGE RECOGNITION

The image recognition was partially validated by splitting the data into a training and testing set in an 80/20 split. The top five accuracy for this model was 90.5%. More thorough validation was completed using K-Folds validation, the industry standard for validating machine learning problems [41]. Multiple models were generated using various 80/20 data splits to confirm that the chosen model's accuracy is maintained regardless of which data is used for training. In each fold, the classifier is trained on the unique training set and tested on the unique testing set. The standard five folds were used with results summarized in Table 6 [42].

Table 6: K-Fold Validation with Five Folds

K-Fold Number	1	2	3	4	5
Top Five Accuracy	92.1%	85.8%	91.3%	89.8%	87.4%

The average accuracy for five folds was 89.3% with a standard deviation of 2.6%. Since the accuracy of the model used in the solution was within half a standard deviation from the average of the five folds, K-folds results strongly suggest that the model accuracy is valid [42].

VII. LIMITATIONS OF DESIGNED SOLUTION

The primary limitations of Canopii are a lack of core features including integration with Kitchener’s database, recording additional traits, updating tree information, and a species identification flow for winter. In order to improve urban forestry, Canopii must integrate with Kitchener’s database such that the information can be used. Currently, there is no integration and Kitchener does not have access to the data collected using Canopii. Another limitation of the current application is that it does not support recording additional traits that are relevant for urban forest benefit analysis such as tree height, DBH and health. There is also no solution for updating information in the database. This means that if the state of a tree changes or if information is incorrectly entered, there is no way for the user to correct it. Lastly, there is currently no manual flow that is viable for tree identification in the winter which limits data collection to only occur in spring and summer.

Another limitation is Canopii’s tree list, as only the 34 species native to Kitchener have an identification flow. Trees not on this list require the users to know the species name in order to use Canopii’s search functionality. This is a severe limitation which prevents expansion beyond Kitchener, as there are 67 trees native to Ontario [23]. Furthermore, the 67 tree species does not include non-native or ornamental trees. In the long term, adding more species to the image processing algorithm will reduce its accuracy. Different classifiers should be trained for each location. However, this is outside of the scope of the project.

The main technical limitation is limited robustness in image recognition. This component only has automatic classification for deciduous trees, which means the user experience for tagging coniferous trees is significantly worse. In addition, Canopii’s image recognition is not reliable enough to have it as a standalone solution; manual identification is still necessary for verification, especially in cases of compound leaves. Furthermore, the training images are only of leaves in summer and do not account for spring, fall and winter variances.

VIII. CONCLUSIONS AND RECOMMENDATIONS

The project concluded with an application that can extract species, location and tree type using automatic and manual methods, and storing results in a central database. The team achieved a top five accuracy of 90.5% for automatic deciduous recognition, with a manual flow available as a backup and for conifers. With these features, the application met objectives as Canopii can successfully engage private residents to provide information on trees in private lands to Kitchener. The design requirements of providing citizens with an easy method of collecting private tree data, creating an accurate automatic tree species recognition system from leaf pictures, and digitizing the manual method of tree identification were successfully met. The project did not meet the objective of integrating with Kitchener.

Canopii can be improved to directly address its limitations. The first improvement would be to integrate the solution with municipalities by exposing the database via an interface or an API endpoint. Another recommendation would be to add functionality for recording tree height, DBH and health ideally with an automated image recognition process. Combined, these two additions would allow Kitchener to utilize the collected tree data to compute local social and environmental benefits by using i-Tree Eco for modelling and analysis [9]. The city will be able to measure and quantify the impact of their urban forest via metrics such as carbon sequestered, urban pollutants reduced and improvement to groundwater flow. Lastly, this data would also inform planning decisions such as new planting locations.

Updating tree information is a critical feature that is planned. Potential solutions include triggering an update flow when adding a tree with the same species and location as an existing tree, allowing users to update their previously tagged trees, or a query and update flow. These options are being explored to improve usability while retaining data integrity.

The final recommendation is to improve image recognition by creating additional classifiers for tree bark, fruit and seeds and obtaining better training data for leaves in summer and fall. This would improve automatic recognition of trees with leaves of low-discernibility, permit tree tagging in all seasons, and allow for automatic recognition of conifers. With these additions, the application would be a solid foundation to improve municipal tree inventories.

ACKNOWLEDGEMENTS

The team would like to thank Igor Ivkovic for providing mentorship, Hamid Tizhoosh for his technical guidance, David Schmitt for his insights into Kitchener’s urban forestry landscape and Matthew Borland for his instruction throughout this course.

REFERENCES

[1] “Kitchener Sustainable Urban Forest Strategy 2019-2039,” Kitchener Urban Forestry. [Online]. Available: https://www.kitchener.ca/en/resourcesGeneral/Documents/INS_PARKS_Sustainable_Urban_Forest_Strategy_DRAFT_SPREAD.pdf. [Accessed: 01-Nov-2019].

[2] “Sustainable Urban Forest Report Card,” City of Kitchener, 2017. [Online]. Available: https://www.kitchener.ca/en/resourcesGeneral/Documents/INS_PARKS_Sustainable_Urban_Forest_Report_Card_2017.pdf. [Accessed: 01-Nov-2019].

[3] “New York City Street Tree Map,” NYC Parks, 2019. [Online]. Available: https://tree-map.nycgovparks.org. [Accessed: 26-Oct-2019].

[4] “ArcGIS Online,” *arcgis.com*. [Online]. Available: https://www.arcgis.com/index.html. [Accessed: 01-Nov-2019].

[5]“PlantSnap,” earth.com, 2019. [Online]. Available: https://www.plantsnap.com. [Accessed: 01-Nov-2019].

[6] N. Kumar, P. N. Belhumeur, A. Biswas, D. W. Jacobs, W. J. Kress, I. C. Lopez, and J. V. B. Soares, “Leafsnap: A Computer Vision System for Automatic Plant Species Identification,” *Computer Vision – ECCV 2012 Lecture Notes in Computer Science*, pp. 502–516, 2012. [Accessed: 01-Nov-2019].

[7] Aerial Forest Inventory System, by J. Lyle. (2011, Jun. 16). *Patent US20130211721A1.* Accessed on: Sep. 26, 2019. [Online]. Available: https://patents.google.com/patent/US9063544. [Accessed: 01-Nov-2019].

[8] Forest Inventory Assessment Using Remote Sensing Data, by Z. Parisa. (2011, Dec. 22). *Patent US9063544B2.* Accessed on: Sep. 26, 2019. [Online]. Available: https://patents.google.com/patent/US20130211721A1/en. [Accessed: 01-Nov-2019].

[9] “i-Tree Eco.” [Online]. Available: https://www.itreetools.org/tools/i-tree-eco. [Accessed: 02-Nov-2019].

[10] D. Bergstrom, K. Rose, J. Olinger, and K. Holley, “The Sustainable Communities Initiative: The Community Engagement Guide for Sustainable Communities,” *Journal of Affordable Housing & Community Development Law*. [Accessed: 01-Nov-2019].

[11] "Climate Change," UN, 2019. [Online]. Available: https://www.un.org/en/sections/issues-depth/climate-change/. [Accessed: 26 Sep 2019].

[12] "Forests, desertification and biodiversity - United Nations Sustainable Development", United Nations Sustainable Development, 2019. [Online]. Available: https://www.un.org/sustainabledevelopment/biodiversity/. [Accessed: 26- Sep- 2019].

[13] “Stormwater to Street Trees, Engineering Urban Forests for Stormwater Management,” Sep-2013. [Online]. Available: https://www.epa.gov/sites/production/files/2015-11/documents/stormwater2streettrees.pdf. [Accessed: 31-Oct-2019].

[14] Agriculture Organization of the United Nations, Urban and Peri-urban Forestry. [Online]. Available: http://www.fao.org/forestry/urbanforestry/87029/en/. [Accessed: 01-Nov-2019].

[15] “Developing a Sustainable Urban Forest Program,” City of Kitchener, 2017. [Online]. Available: https://www.kitchener.ca/en/resourcesGeneral/Documents/INS_OPS_UrbanForestry_Developing-a-sustainable-urban-forest-program.pdf. [Accessed: 31 Oct 2019].

[16] J. Nielsen, "Iterative user-interface design," in Computer, vol. 26, no. 11, pp. 32-41, Nov. 1993. [Accessed: 01-Nov-2019].

[17] Building the Analysis Object Model. [Online]. Available: http://www.cs.sjsu.edu/~pearce/modules/lectures/ooa/analysis/AnalysisObjectModel.htm. [Accessed: 19-Mar-2020].

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” Communications of the ACM, vol. 60, no. 6, pp. 84–90, 2017.

[19] “ImageNet Leaderboard: Papers with Code,” Papers With Code : the latest in machine learning. [Online]. Available: https://paperswithcode.com/sota/image-classification-on-imagenet. [Accessed: 19-Mar-2020].

[20] A. K. Reyes, J. C. Caicedo, and J. E. Camargo, “Fine-tuning Deep Convolutional Networks for Plant Recognition,” Laboratory for Advanced Computational Science and Engineering Research. [Accessed: 01-Nov-2019].

[21] M. M. Ghazi, B. Yanikoglu, and E. Aptoula, “Plant identification using deep neural networks via optimization of transfer learning parameters,” Neurocomputing, vol. 235, pp. 228–235, 2017. [Accessed: 01-Nov-2019].

[22] H. Esmaeili and T. Phoka, “Transfer Learning for Leaf Classification with Convolutional Neural Networks,” 2018 15th International Joint Conference on Computer Science and Software Engineering (JCSSE), 2018. [Accessed: 01-Nov-2019].

[23] “The Tree Atlas,” Ontario.ca. [Online]. Available: https://www.ontario.ca/environment-and-energy/tree-atlas/. [Accessed: 01-Nov-2019].

[24] D. Schmitt, private communication, Jan. 2020.

[25] “iNaturalist,” iNaturalist. [Online]. Available: https://www.inaturalist.org/. [Accessed: 2-Feb-2020].

[26] P. Lameski, E. Zdravevski, R. Mingov, and A. Kulakov, “SVM Parameter Tuning with Grid Search and Its Impact on Reduction of Model Over-fitting,” Lecture Notes in Computer Science Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, pp. 464–474, 2015.

[27] J. L. Farrar, Trees in Canada. Ottawa: Natural Resources Canada, Canadian Forest Service, 2019.

[28] L. J. Kershaw, Trees of Ontario: including tall shrubs. Edmonton: Lone Pine Pub., 2001.

[29] J. D. Pokorny, Urban tree risk management: a community guide to program design and implementation. St. Paul, MN: USDA Forest Service, Northeastern Area, State and Private Forestry, 2003.

[30] “Executive Summary,” CCR2019, 2019. [Online]. Available: https://changingclimate.ca/CCCR2019/chapter/executive-summary/. [Accessed: 19 Mar 2020].

[31] “The Cost of Climate Adaptation,” Insurance Bureau of Canada. [Online]. Available: http://www.ibc.ca/on/disaster/water/flooding-in-canada/the-cost-of-climate-adaptation. [Accessed: 19-Mar-2020].

[32] K. H. Clark and K. A. Nicholas, “Introducing urban food forestry: a multifunctional approach to increase food security and provide ecosystem services,” Landscape Ecology, vol. 28, no. 9, pp. 1649–1669, 2013.

[33] “Green jobs grow in the forest,” Green jobs grow in the forest. [Online]. Available: http://www.fao.org/europe/news/detail-news/en/c/1152945/. [Accessed: 19-Mar-2020].

[34] A. P. Kythreotis, C. Mantyka-Pringle, T. G. Mercer, L. E. Whitmarsh, A. Corner, J. Paavola, C. Chambers, B. A. Miller, and N. Castree, “Citizen Social Science for More Integrative and Effective Climate Action: A Science-Policy Perspective,” *Frontiers in Environmental Science*, vol. 7, May 2019.

[35] “State · React Native,” React Native. [Online]. Available: https://reactnative.dev/docs/state. [Accessed: 19-Mar-2020].

[36]“Redux - A predictable state container for JavaScript apps.: Redux.” [Online]. Available: https://redux.js.org/. [Accessed: 19-Mar-2020].

[37] D. R. Morse and G. M. Tardivel, “A Comparison of the Effectiveness of a Dichotomous Key and ...” [Online]. Available: https://kar.kent.ac.uk/21343/1/WoodliceMorse.pdf. [Accessed: 19-Mar-2020].

[38] “Documentation” *Django*. [Online]. Available: https://docs.djangoproject.com/en/3.0/. [Accessed: 19-Mar-2020].

[39] J. Nielsen, "Iterative user-interface design," in Computer, vol. 26, no. 11, pp. 32-41, Nov. 1993. [Accessed: 01-Nov-2019].

[40] M. Ivanković, G. Petrović, R. Just, and G. Fraser, “Code coverage at Google,” Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering - ESEC/FSE 2019, 2019.

[41] M. Kaariainen, “Semi-Supervised Model Selection Based on Cross-Validation,” The 2006 IEEE International Joint Conference on Neural Network Proceedings.

[42] J. D. Rodriguez, A. Perez and J. A. Lozano, "Sensitivity Analysis of k-Fold Cross Validation in Prediction Error Estimation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 3, pp. 569-575, March 2010.

TEAM MEMBER CONTRIBUTIONS TO ENGINEERING ANALYSIS & DESIGN

Contributions to Team Project Engineering Analysis and Design		
Team Member	**Description of Contributions, including Tools used.**	**Outcomes**
Ethan Liang	● Architected and implemented the back-end in Docker including server with Django and database with PostgreSQL ● Worked with Kevin to create communication pipelines between device and back-end ● Developed server infrastructure on AWS and performed the team's DevOps work ● Worked with Derin to integrate image recognition onto Django server ● Worked with Kevin and Derin to collect and manually vet images for training and testing	● Developed the infrastructure supporting the communication between the different parts of the project ● Created the database and back-end to process and store all data received from app users
Derin Denizkusu	● Created machine learning infrastructure through Keras for project ● Worked on creating leaf image classification system as well as researching and planning complete image recognition solution ● Created software architecture design for complete solution ● Manually vetted and edited new images, cropping and rotating ~600 leaf images to fit image recognition requirements ● Validated machine learning with k-folds ● Conducted user testing prior to final prototype using Figma and powerpoint slides ● Worked with Jessie and Kevin on various descriptions, images and flow content for phone application	● Image recognition is functional with a 90.5% top five accuracy ● Image recognition has been validated with k-folds ● Preliminary user test data collected ● Descriptions and content are in the application
Kevin Luong	● Conducted software engineering analysis to determine entity, control and boundary objects ● Designed and created a stateful mobile application using React-Native and Redux ● Wrote unit tests for the application ● Worked with Jessie to lead meeting planning and assigned tasks to teammates (Trello, Google Docs) ● Worked with Ethan to integrate the application and server ● Manually vet iNaturalist images ● Conducted final round user testing using the application	● Cross platform mobile application which is integrated with server and database ● Created demos and prototype review videos
Jessie Won	● Owned and led all design-related tasks, including UI designs, prototypes, diagrams,	● Created all wireframes, mockups, and prototypes which

	tables, and illustrations (Figma, Google Docs, Google Sheets) • Worked with Kevin to lead meeting planning and assigned tasks to teammates (Trello, Google Docs) • Conducted in-person usability testing using both the product and prototypes (Figma) • Worked with Kevin on final front-end engineering tweaks, mainly UI and visual changes (React-Native)	were used for testing and development • Created all diagrams (ex. system diagram), tables, and illustrations (ex. tree trait illustrations, logos, icons) • Created symposium poster • Gathered results from testing that were integrated into the next iteration designs