Analyse informatique des images biologiques Réseaux neuronaux pour le traitement d'images

Dirk Hillmer

université de BORDEAUX

THÈSE PRÉSENTÉE

POUR OBTENIR LE GRADE DE

DOCTEUR DE

L'UNIVERSITÉ DE BORDEAUX

ÉCOLE DOCTORALE Sciences de la Vie et de la Santé (SVS)
SPÉCIALITÉ Biologie Cellulaire et Physiopathologie

Par Dirk Hillmer

Analyse informatique des images biologiques
Réseaux neuronaux pour le traitement d'images

Sous la direction de: Prof. Martin HAGEDORN

Soutenue le 27.05.2024

Membres du jury:

M. Mulle, Christophe, Professor Bordeaux Neurocampus , Président du jury
M. Hagedorn, Martin Professor Bordeaux Neurocampus Directeur de thèse
Mme. Orend, Gertraud Directrice de recherche Inserm Université de Strasbourg, Rapporteuse
M. Wilting, Jörg Professeur Université Göttingen Rapporteur

Titre: Analyse informatique des images biologiques
Réseaux neuronaux pour le traitement d'images

Résumé: L'IA en médecine est un domaine en croissance rapide et son importance en dermatologie est de plus en plus prononcée. Les progrès des réseaux neuronaux, accélérés par de puissants Unité de Traitement Graphique, ont catalysé le développement de systèmes d'IA pour l'analyse des troubles cutanés. Cette étude présente une nouvelle approche qui exploite les techniques d'infographie pour créer des réseaux d'IA adaptés aux troubles cutanés. La synergie de ces techniques génère non seulement des données de formation, mais optimise également la manipulation des images pour un traitement amélioré. Le vitiligo, un trouble cutané dépigmentant courant, constitue une étude de cas poignante. L'évolution des thérapies ciblées souligne la nécessité d'une évaluation précise de la surface touchée. Cependant, les méthodes d'évaluation traditionnelles prennent beaucoup de temps et sont sujettes à une variabilité inter-évaluateur et intra-évaluateur. En réponse, cette recherche vise à construire un système d'intelligence artificielle (IA) capable de quantifier objectivement la gravité du vitiligo facial. La formation et la validation du modèle d'IA ont exploité un ensemble de données d'une centaine d'images de vitiligo facial. Par la suite, un ensemble de données indépendant de soixante-neuf images de vitiligo facial a été utilisé pour l'évaluation finale. Les scores attribués par trois médecins experts ont été comparés aux performances inter-évaluateurs et intra-évaluateurs, ainsi qu'aux évaluations de l'IA. De manière impressionnante, le modèle d'IA a atteint une précision remarquable de 93 %, démontrant son efficacité dans la quantification de la gravité du vitiligo facial. Les résultats ont mis en évidence une concordance substantielle entre les scores générés par l'IA et ceux fournis par les évaluateurs humains. Au-delà du vitiligo facial, l'utilité de ce modèle dans l'analyse des images du corps entier et des images sous différents angles est apparue comme une voie d'exploration prometteuse. L'intégration de ces images dans une représentation complète pourrait offrir un aperçu de la progression du vitiligo au fil du temps, améliorant ainsi le diagnostic clinique et les résultats de la recherche. Bien que le voyage ait été fructueux, certains aspects de la recherche se sont heurtés à des obstacles en raison de ressources insuffisantes en images et en données. Une exploration de l'analyse de modèles de souris in vivo et de l'analyse de la pigmentation des cellules de la peau dans des modèles d'embryons précliniques ainsi que de la reconnaissance d'images de la rétine a malheureusement été interrompue. Néanmoins, ces défis mettent en lumière la nature dynamique de la recherche et soulignent l'importance de l'adaptabilité pour surmonter les obstacles imprévus.
En conclusion, cette étude met en valeur le potentiel de l'IA pour révolutionner l'évaluation dermatologique. En fournissant une évaluation objective de la gravité du vitiligo facial, le modèle d'IA proposé constitue un complément précieux à l'évaluation humaine, tant dans la pratique clinique que dans la recherche. La poursuite continue de l'intégration de l'IA dans l'analyse de divers ensembles de données d'images est prometteuse pour des applications plus larges en dermatologie et au-delà.

Mots clés: Vitiligo, Artificial Intelligence, Image Processing, Computer Graphics

Title: Computer-based Analysis of Biological Images Neuronal Networks for Image Processing

Abstract: AI in medicine is a rapidly growing field, and its significance in dermatology is increasingly pronounced. Advancements in neural networks, accelerated by powerful Graphics Processing Units, have catalyzed the development of AI systems for skin disorder analysis. This study presents a novel approach that harnesses computer graphics techniques to create AI networks tailored to skin disorders. The synergy of these techniques not only generates training data but also optimizes image manipulation for enhanced processing. Vitiligo, a common depigmenting skin disorder, serves as a poignant case study. The evolution of targeted therapies underscores the necessity for precise assessment of the affected surface area. However, traditional evaluation methods are time-intensive and prone to inter- and intra-rater variability. In response, this research endeavors to construct an artificial intelligence (AI) system capable of objectively quantifying facial vitiligo severity. The AI model's training and validation leveraged a dataset of one hundred facial vitiligo images. Subsequently, an independent dataset of sixty-nine facial vitiligo images was used for final evaluation. The scores assigned by three expert physicians were compared with both inter- and intra-rater performances, as well as the AI's assessments. Impressively, the AI model achieved a remarkable accuracy of 93%, demonstrating its efficacy in quantifying facial vitiligo severity. The outcomes highlighted substantial concordance between AI-generated scores and those provided by human raters. Expanding beyond facial vitiligo, this model's utility in analyzing full-body images and images from various angles emerged as a promising avenue for exploration. Integrating these images into a comprehensive representation could offer insights into vitiligo's progression over time, thereby enhancing clinical diagnosis and research outcomes. While the journey has been fruitful, certain aspects of the research encountered roadblocks due to insufficient image and data resources. An exploration into the analysis of in vivo mouse models and analyzing pigmentation of skin cells in preclinical embryo models as well as retina image recognition was regrettably halted. Nevertheless, these challenges illuminate the dynamic nature of research and underscore the importance of adaptability in navigating unforeseen obstacles.
In conclusion, this study showcases the potential of AI to revolutionize dermatological assessment. By providing an objective evaluation of facial vitiligo severity, the proposed AI model offers a valuable adjunct to human assessment in both clinical practice and research settings. The ongoing pursuit of integrating AI into the analysis of diverse image datasets holds promise for broader applications in dermatology and beyond.

Keywords: Vitiligo, Artificial Intelligence, Image Processing, Computer Graphics

Unité de recherche

[BRIC , INSERM UMR1312, Team 5, University of Bordeaux; Dermatology Department,
CNRS UMR 5164 ImmunoConCept, équipe 2, groupe Immuno-Dermatologie
University Hospital of Bordeaux,
Bordeaux, France. 351 cours de la Libération CS10004 33405 Talence CEDEX]

Table of Contents

Table of Figures

List of Abbreviations

PhD - Doctor of Philosophy
PPT - PowerPoint Presentation
AI - Artificial Intelligence
CV - Computer Vision
UNet - UNet (a convolutional neural network architecture)
OpenCV - Open Source Computer Vision Library
JPG - Joint Photographic Experts Group
PNG - Portable Network Graphics
RGB - Red, Green, Blue
HSV - Hue, Saturation, Value
YCrCb - Luminance (Y), Chrominance (Cr), and Chrominance (Cb)
TIFF - Tagged Image File Format
CV2 - OpenCV version 2. OpenCV (Open Source Computer Vision Library)
ACO - Ant Colony Optimization
PyACO - Python implementation of the Ant Colony Optimization algorithm
CNN - Convolutional Neural Network
ORB - Oriented FAST and Rotated BRIEF (an algorithm for keypoint detection)
Flask - A micro web framework written in Python
MySql - MySQL (a relational database management system)
Tensorflow - TensorFlow (an open-source machine learning framework developed by Google)
SLURM - Simple Linux Utility for Resource Management
3D - Three-dimensional
BLOG - Weblog or online journal
DR - Doctor or Clinician
API - Application Programming Interface
PDF - Portable Document Format
SVM - Support Vector Machines
CLAHE - Contrast Limited Adaptive Histogram Equalization
Epochs - One complete pass through the entire dataset during the training process
Batch Size - The number of training examples processed in one iteration
Dataset - The collection of input-output pairs used to train and evaluate the neural network.
Activation Function - Introduces non-linearity to the output of a neuron in a neural network.
Neuron - A fundamental unit of a neural network, that receives input, performs a computation, and produces an output.
Neural Network - A network composed of interconnected neurons that process input data to produce output predictions.
ReLU: Rectified Linear Unit - An activation function commonly used in neural

networks to introduce non-linearity.
Sigmoid Function - An activation function that maps input values to a range between 0 and 1 for binary classification tasks.
Dice Coefficient - A similarity metric used to measure the agreement between two sets, commonly used in image segmentation tasks.
Jaccard Index - Also known as Intersection over Union (IoU), a measure of the overlap between two sets, often used to evaluate segmentation tasks.

Introduction

The thesis is focused on the analysis of biological and medical images in the field of dermatology or related disciplines where sufficient amounts of image data are available. Not only the amount of data is important also the quality of the data is a major success factor. The clinical images taken are suitable for the clinical analysis but in some instances are not suitable for AI processing. In general, medical trained person's visual inspection of image data regardless of the source is misleading, and important details are overseen because of restricted analytical capacities. Nevertheless, the experience of a medically trained person has also an advantage over machine learning since images in difficult circumstances can be analyzed with more background information than an AI can have. Recent advances in the use of artificial intelligence (AI) [1], machine learning, and neural networks have provided the scientific community with unprecedented analytical tools [2]. Implementation of such tools in a concrete medical or biological framework remains a challenge since it requires tight collaboration of specialists in the field of informatics with AI programming capacities, biologists or physicians, and bioinformaticians as well access to biological and/or medical image data [3] [4] [5].

The proposed thesis project inserts into this context and is mainly based on three thematic aims:

1. Development of powerful analytical tools to characterize and quantify alterations of the vascular tree in vivo using mouse models
2. Analyzing pigmentation of skin cells in a preclinical embryo model (Xenopus Laevis).
3. Validation of algorithms in a large number of clinical dermatological images, and data in pathologies related to the first two items.
4. Analysing Vitiligo images to validate the therapeutic improvement.

The goal is to generate robust strategies that could be patented and applied to concrete clinical workflow to ameliorate complex clinical diagnosis and therapeutic decision-making [7] without complicating the clinical diagnosis with too many restrictions or boundaries.

The idea to use computers in the medical sector was already started decades ago. The programming language Prolog was one of the first to contain concepts of AI[6]. The lack of proof that XOR(exclusive or) can be used made the use and progress of this area less relevant. The advantages of computer vision arose in the 80s when more sophisticated graphic cards and computers with more calculation power were created. This leads to some algorithms and formats that are targeted for image processing with machines. The way machines and persons conduct image analyses

are different since the person can always rely on experience. The paradigm of algorithms systematically describing a problem evolved to the combinatorial calculation of states a system is in. Predictions of the outcome in a non-deterministic calculation were the next stage. Quantum computing is the advanced state of this sort of prediction. The states are connected networks that allow them to operate on a reduced set of data to predict outcomes. The advantage of those networks is the possibility to reduce the complexity of normal algorithms where there is a need to define all possibilities to get a reliable result [8]. The difficulty is to find all the conditions to create a functional algorithm a lot of cases need to be specified. The opposite approach to using a combinatorial network that creates weights for the transition of data is a more uncontrollable way which in return provides a predicted answer. The information we can put into such a network of states is used for training and validation. This means the input data needs to be of the same kind. The information can not change dramatically for the network to perform properly. A certain number of data needs to be available for training such a network. The connections need to be set and are measured by the use of validation data. The following section will discuss some of the relevant computer vision and AI algorithms concerning the image recognition and object detection needed.

Material and Methods

Understanding Key Concepts in AI

To work effectively with AI, it is essential to understand key terms and concepts that form the foundation of AI. The following section elucidates the most important concepts necessary for navigating the AI landscape:

1. Dataset:
Definition: The dataset encompasses the collection of input-output pairs used for training and evaluation.

Influence: Dataset quality, size, and diversity directly impact model performance and generalization.

Impact on Training: A diverse and representative dataset enables the model to learn robust features and patterns, enhancing performance on unseen data.

2. Epochs:
Definition: An epoch refers to a single pass through the entire dataset during the training process of a neural network.

Influence: The number of epochs determines how many times the neural network processes the entire dataset, impacting model convergence and performance.

Impact on Training: While increasing epochs may enhance convergence and performance, excessive epochs can lead to overfitting, hindering the model's ability to generalize to unseen data.

3. Batch Size:
Definition: Batch size denotes the number of training examples processed in one iteration by the neural network.

Influence: Batch size affects training speed and stability, with larger batches typically resulting in faster training but requiring more memory.

Impact on Training: Smaller batch sizes facilitate more frequent model updates and better generalization, whereas larger batches may expedite convergence but risk poorer generalization.

4. Activation Function:
Definition: An activation function introduces non-linearity to a neuron's output in a neural network.

Influence: Activation functions determine neuron output range and facilitate learning complex relationships between features.

Impact on Training: Selecting the appropriate activation function is vital for effective training, with common options including ReLU and sigmoid, each suited to different scenarios based on desired output behavior.

5. Neuron:
Definition: A neuron is a basic unit of computation in a neural network, receiving inputs, applying weights, and producing an output.

Influence: Neurons process information and contribute to the network's ability to learn and make predictions.

Impact on Training: Proper configuration and arrangement of neurons within layers are essential for achieving desired network behavior and performance.

6. Neural Network:
Definition: A neural network is a computational model inspired by the structure and function of the human brain, consisting of interconnected neurons organized into layers.

Influence: Neural networks can learn complex patterns and relationships from data, making them powerful tools for tasks such as classification, regression, and pattern recognition.

Impact on Training: The architecture, depth, and connectivity of neural networks influence their ability to effectively represent and learn from data.

7. ReLU (Rectified Linear Unit):
Definition: ReLU is an activation function that introduces non-linearity by outputting the input directly if it is positive; otherwise, it outputs zero.

Influence: ReLU addresses the vanishing gradient problem and accelerates training convergence in deep neural networks.

Impact on Training: ReLU activation is widely used in neural network architectures due to its simplicity and effectiveness in improving model performance.

8. Sigmoid Function:

Definition: The sigmoid function is a mathematical function that maps input values to a range between 0 and 1, providing non-linearity to neural network outputs.

Influence: Sigmoid functions are commonly used in binary classification tasks, where they squash arbitrary input values to a probability range.

Impact on Training: Despite being prone to vanishing gradients, sigmoid functions remain valuable for tasks requiring probability outputs or where the output space is constrained.

9. Dice Coefficient:

Definition: The Dice coefficient is a similarity measure commonly used in image segmentation tasks to quantify the agreement between predicted and ground truth segmentations.

Influence: The Dice coefficient assesses segmentation accuracy by computing the overlap between predicted and actual segmentations.

Impact on Training: Maximizing the Dice coefficient during training encourages the model to produce segmentations that closely align with ground truth annotations, enhancing overall performance.

10. Jaccard Index:

Definition: The Jaccard index, also known as the intersection over union (IoU), is a similarity measure used to evaluate the overlap between two sets.

Influence: In image segmentation tasks, the Jaccard index quantifies the agreement between predicted and ground truth segmentations.

Impact on Training: Maximizing the Jaccard index encourages the model to generate segmentations that accurately delineate object boundaries, leading to improved segmentation quality.

With these fundamental terms clarified, we embark on a journey into the realm of AI. We will use these terms in the following sections and other terms used in the computer graphics area.

Analyzing pigmentation of skin cells in a preclinical embryo model (Xenopus Laevis)

The objective of this study is to analyze the pigmentation of skin cells in a preclinical embryo model (Xenopus Laevis) [58] using a combination of imaging techniques and artificial intelligence algorithms.

The analysis of pigmentation in such models provides valuable insights into biological processes and can contribute to the understanding of dermatological conditions and treatments.

Analyzing Pigmentation in Preclinical Embryo Models:

To investigate pigmentation in preclinical embryo models, various imaging techniques can be employed. These include confocal microscopy, fluorescence microscopy, and brightfield microscopy [9]. Each technique offers distinct advantages and enables different aspects of pigmentation analysis [10].

Confocal microscopy is a powerful tool for analyzing pigmentation in preclinical embryo models. It utilizes laser excitation of fluorescent dyes or proteins within skin cells, allowing for the generation of high-resolution, three-dimensional images. This technique enables precise visualization of pigmentation distribution, intensity, and spatial relationships between cells [11].

Fluorescence microscopy can also be used to analyze pigmentation by labeling skin cells with fluorescent dyes or proteins that specifically bind to pigments like melanin [12]. This method provides high-contrast images, facilitating the visualization and quantification of pigmented cells [13].

Brightfield microscopy, a simpler and more accessible imaging technique, employs white light illumination to visualize pigmented skin cells in preclinical embryo models. By identifying differences in light absorption and reflection between pigmented and non-pigmented cells, this technique allows for their visualization. However, it may not provide the same level of detail and sensitivity as confocal or fluorescence microscopy [14].

Image analysis software plays a vital role in quantifying pigmentation in preclinical embryo models. By analyzing the intensity, size, and distribution of pigmented regions within images, the software can provide valuable insights and comparisons to reference

standards or control samples. Statistical analysis can further determine significant differences in pigmentation between experimental groups or treatments. Notably, the field of Systematic Biology offers relevant insights in this regard [15].

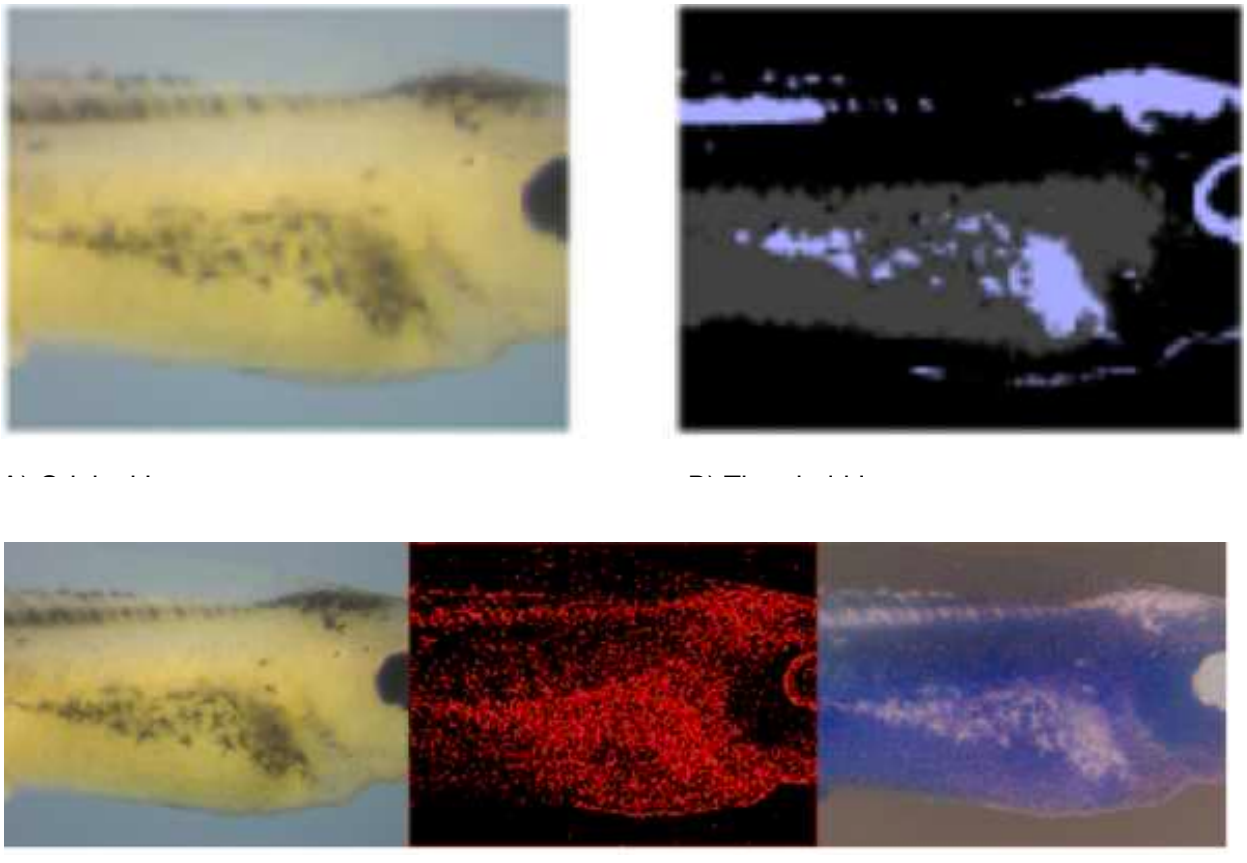
Image Analysis and AI Methods:

AI algorithms, particularly convolutional neural networks (CNNs), can be employed to train deep learning models for pigmentation analysis. This requires a large dataset of annotated images, encompassing preclinical embryo models with labeled and segmented pigmented regions. Once trained, CNNs can automatically segment pigmented regions in new images, offering a more objective and consistent approach compared to manual methods [16].

Additional AI Techniques

To create a mask we applied thresholding to the images. With the contained masks we run a vector graphic algorithm that creates lines around the objects found. With the PyPoTrace (https://pypi.org/project/pypotrace/) we created patterns around the masks created with thresholding. With an overlay, we enhanced the original image with the mask information. As a result, we were able to present the Original Image 1C) the mask 1D) and the overlay of the mask and image in 1E).

Figure 1) Embryo Models Analysis



C) Original Image D) Mask E) Image with overlay

Introduction to Retina Image Recognition

The field of retina image recognition research is extensive, with ongoing studies focusing on the analysis of the retina and its relation to diabetic diseases[19]. The precise localization of specific information within the retina is crucial for accurate analysis and diagnosis.

Techniques in Retina Recognition

Retina image recognition involves the application of machine learning and computer vision techniques to analyze and classify medical images of the retina. Retinal images, obtained using specialized cameras, play a vital role in diagnosing and monitoring various eye diseases, including age-related macular degeneration, diabetic retinopathy, and glaucoma.

AI and Machine Learning Training

Convolutional neural networks (CNNs)[20], a type of deep learning algorithm well-suited for image analysis tasks, are commonly employed in retina image recognition[21]. CNNs can be trained on large datasets of labeled retinal images, enabling them to learn patterns and features associated with different retinal diseases. Once trained, CNNs can automatically analyze new retinal images and classify them based on the presence or severity of various eye diseases[22]. This assists healthcare professionals in making accurate diagnoses and informed treatment decisions.

Image Processing Techniques

Other machine-learning techniques, such as support vector machines and random forests, can also be used for retina image recognition[23] [24]. Additionally, specialized image processing techniques, such as contrast enhancement, noise reduction, and feature extraction, can be applied to improve the accuracy of classification results[25].

Support Vector Machines (SVMs)

Support Vector Machines are a class of supervised machine learning algorithms used for classification and regression tasks[53]. They are particularly effective for binary classification but can be extended to multi-class problems as well.

In SVM, the algorithm aims to find a hyperplane that best separates data points belonging to different classes. This hyperplane is referred to as the "decision boundary." The key idea behind SVM is to maximize the margin, which is the distance between the decision boundary and the nearest data points of each class. These nearest data points are known as "support vectors," hence the name Support Vector Machines.

SVMs are known for their ability to handle high-dimensional data and are effective in cases where the data is not linearly separable. To address this, they can employ techniques such as kernel functions (e.g., polynomial or radial basis function kernels) to map the data into a higher-dimensional space, where linear separation becomes possible[54].

Random Forests
Random Forests are an ensemble learning technique used for both classification and regression tasks [63]. They are built on the foundation of decision trees. However, instead of relying on a single decision tree, Random Forests combine the results of multiple decision trees to make more robust predictions.

In a Random Forest, a collection of decision trees is generated[54], each trained on a random subset of the data (bootstrap samples) and a random subset of the features (feature bagging). During the prediction phase, each tree "votes" for the class (in classification) or produces a prediction (in regression), and the final prediction is determined by majority voting (in classification) or averaging (in regression).

Random Forests are known for their high accuracy, robustness against overfitting, and ability to handle large datasets with high dimensionality. They are also less prone to the problem of decision tree bias and variance, making them a popular choice for various machine learning tasks[55].

In summary, SVMs are effective for finding decision boundaries that maximize the margin between classes, while Random Forests harness the collective wisdom of multiple decision trees to enhance prediction accuracy and generalization. Both are powerful machine learning algorithms used in different contexts based on the nature of the data and the problem at hand.

Existing Resources in Retina Image Recognition
For retina and image recognition, extensive research has already been conducted[26][27], and databases containing annotated datasets and AI models for retina images are available. In the realm of retina image recognition, a wealth of invaluable resources and databases has been cultivated, serving as essential pillars for research and development in this critical domain. These repositories not only offer extensive datasets but also exhibit distinct characteristics that render them indispensable for the scientific community[28][29].

1. **The Kaggle Retina Database:** Among the most prominent resources is the Kaggle Retina Database, a colossal collection of over 100,000 high-resolution retinal images sourced from diverse clinical studies and healthcare institutions worldwide. This dataset encompasses a comprehensive spectrum of retinal conditions, including age-related macular degeneration, diabetic retinopathy, and glaucoma, making it an exceptional asset for training and validating deep learning models.

2. **The Diabetic Retinopathy Dataset:** Recognized for its specificity, the Diabetic Retinopathy Dataset focuses exclusively on diabetic retinopathy, comprising 20,000 meticulously annotated retinal images. These images are meticulously labeled according to the severity of diabetic retinopathy, enabling researchers to develop models tailored to this prevalent condition.

3. **The DEF Glaucoma Image Repository:** For investigations into glaucoma, the DEF Glaucoma Image Repository stands as an indispensable resource. It offers a substantial collection of 30,000 glaucomatous and healthy retinal scans, acquired through cutting-edge imaging technologies. The repository also includes detailed patient demographics, facilitating comprehensive analyses of demographic influences on disease progression.

4. **The GHI AI Model for Retina Analysis:** In addition to datasets, the GHI AI Model for Retina Analysis has garnered widespread acclaim. This pre-trained convolutional neural network (CNN) is renowned for its exceptional accuracy in classifying retinal conditions. Trained on a diverse range of retinal datasets, this model has become a benchmark for researchers aiming to harness the power of deep learning in their studies.

The image set provided contained 5 images of the retina. The data was analyzed with thresholding to find a way to automatically generate masks for AI training.

Introduction to In Vivo Mouse Models

The Analysis of in vivo mouse models involves the use of computational and analytical methods to study the biology of living mice in their natural environment. In vivo mouse models are extensively employed in biomedical research to investigate various biological processes, including disease development, drug discovery, and the effects of genetic mutations.

Imaging Techniques in In-Vivo Mouse Models

Advanced imaging techniques, such as confocal microscopy, can be used to observe and track the movement of cells or specific molecules in live mice, providing insights into cellular behavior and disease progression [30]. Bioinformatics and data analysis tools are also utilized to analyze large datasets generated from in vivo experiments. Machine learning algorithms, such as neural networks and support vector machines, can identify patterns and correlations in complex datasets, aiding in the development of predictive models for disease progression and treatment response [31].

Microbiome Studies in In Vivo Mouse Models

In vivo mouse models are also instrumental in studying the interactions between the microbiome and the host, and how these interactions contribute to disease. High-throughput sequencing technologies can be used to profile the microbiome, and bioinformatics tools can identify microbial species and their functional characteristics [32].

Digital Pathology and AI Techniques

Digital pathology, including the use of whole-slide imaging (WSI) technology, offers new opportunities in pathology by enabling remote primary diagnostics, workload efficiency, collaborations, teleconsultation, central clinical trial review, image analysis, virtual education, and innovative research. Computer vision and artificial intelligence (AI) techniques, including deep learning algorithms and image recognition, have further expanded the possibilities of computational pathology [33].

Artificial neural networks (ANNs), particularly those utilizing deep learning[61], have made significant progress in computational pathology. Training deep learning networks requires a substantial amount of high-quality data. The quantity and quality of data are crucial factors for successful model training [34].

Blood Vessel Analysis in In Vivo Mouse Models

The analysis of mouse ears for cancer prediction is a manual process that could be automated using computer vision and AI. Initial steps involve data analysis and segmentation, separating the foreground (blood vessels) from the background. With

algorithms inspired by human context and memory, computer graphics methods can be applied to discover and analyze images [35].

The location of blood vessels can be achieved by applying a threshold and calculating the background and foreground information. Traced vector graphics generated using libraries like PyPoTrace can be overlaid onto in vivo mouse model images Figure 2), aiding in the visualization and analysis of blood vessels [36][37].
Figure 2) Mice Ear Analysis

An alternative method was developed to obtain ground truth data for the ear analysis. This involved the creation of a tool to extend and separate image channels, coupled with adjustments and image graphic algorithms aimed at enhancing the images to enable the extraction of blood vessels. The mask generation process entailed splitting the image and its corresponding mask into patches of size 128 x 128. Subsequently, this data was utilized to initiate the training of a neural network using a UNet architecture, specifically an Attention Residual UNet as depicted in Figure 2-1. Additionally, the images were enriched using a contrast-limited Adaptive Histogram Equalization (CLAHE) algorithm, which enhances contrast to facilitate better analysis. Research on blood vessel detection has already been conducted [66]. The use of a hessian matrix to visualize the blood vessels in MRT scan images or other images.

Introduction to Vitiligo Research

Vitiligo is a skin condition characterized by the loss of pigmentation in patches on the skin[38]. This research aims to develop a system that can calculate the extent of vitiligo in an image, indicate the affected area, and provide a measure of the condition. In recent years, there has been a growing focus on vitiligo research due to the development of new therapies that necessitate measurable results. Systems for skin cancer detection of dark spots have already been developed [64]. A rich source of patient data in the form of photographs was provided by the University of Bordeaux Dermatology.

Vitiligo is defined by the presence of white patches on the skin, which can appear anywhere on the body, but are commonly found on the face, hands, arms, and feet. The loss of pigmentation occurs due to the destruction of melanocytes, the cells responsible for producing melanin—the pigment that gives color to the skin, hair, and eyes.

The exact cause of vitiligo is not fully understood, although it is believed to be an autoimmune disorder. In this condition, the body's immune system mistakenly attacks and destroys its melanocytes. Genetic predisposition, exposure to environmental triggers, and oxidative stress are among the factors that may contribute to the development of vitiligo.

It is important to note that vitiligo is not contagious and does not typically cause physical harm. However, its impact on individuals can be significant, particularly in cultures where fair skin is highly valued, leading to psychological and social challenges. While there is currently no known cure for vitiligo, various treatments such as phototherapy, topical corticosteroids, and skin grafting can help restore some of the lost pigmentation and improve the appearance of the skin [39].

Clinical Manifestations and Diagnosis

The clinical manifestations of vitiligo, its localization, the time of Occurrence, and the course of the disease are very different. Usually, the Vitiligo is visually diagnosed and the efflorescences are typical in their appearance and lack natural skin hormones. The patches of vitiligo are typically white and have well-defined borders, which may be irregular or smooth. The patches may occur in any part of the body but are most commonly found on the face, hands, arms, and feet. Vitiligo can also affect mucous membranes, such as the mouth and genitals. This is one feature that might be of use for recognition.

The time of onset and course of the disease can also vary widely. Vitiligo can develop at any age, but it most commonly begins in childhood or young adulthood.

The disease may progress slowly over many years or may occur rapidly, with new patches appearing quickly.

In some cases, vitiligo may remain stable for long periods, while in other cases, the patches may continue to spread and enlarge. In rare cases, vitiligo may spontaneously regress, with the affected areas returning to their normal pigmentation.

Overall, the clinical manifestations of vitiligo can have a significant impact on a patient's quality of life, particularly in terms of their psychological and social well-being. Treatment options for vitiligo include topical and systemic medications, phototherapy, and surgical approaches such as skin grafting [6].

Clinic Image
Vitiligo can occur on any region of the skin that contains pigment, including the face, neck, arms, hands, legs, feet, and even the scalp or genital area. It can also affect mucous membranes such as the inside of the mouth or nose. The distribution and severity of vitiligo can vary widely from person to person. In some cases, the affected areas may be small and localized, while in others, large areas of the body may be affected.

The typical efflorescences are depigmented areas of different types sizes, shapes, and numbers on the skin, which impress with a mostly milky white well-defined border, which may also be hyper-pigmented. The shape of spots is round to oval, and the distribution is often bilaterally symmetrical [7].

The skin field in the affected areas is preserved [2, 8]. Rarely rapidly progressing forms occur. These are also called Harlequin referred to as vitiligo.
Predilection sites are predominantly areas with increased stress due to friction and movement [7], as well as through exposure to the sun.
The hair in the areas of vitiligo can be both normally pigmented be, as well as grow pigment-free. There are four forms of Vitiligo categories:

• Trichrome Vitiligo: A central depigmented zone is surrounded by a medium-brown, not yet completely depigmented border and neighboring, normally pigmented skin

• Quadrichrome Vitiligo: In quadrichrome Vitiligo, perifollicular-lare or rim-accentuated foci with hyperpigmentation, which are often found in the is seen as part of re-pigmentation processes.

• Confetti spots: they have the typical color of vitiligo, but are only 1 to 2 mm in diameter and appear anywhere or perifollicular.

• Inflammatory Vitiligo: A central depigmented zone is surrounded by an erythematous, raised rim

Those variations of vitiligo make it difficult to create a fixed algorithm for detection. One type of vitiligo has sharp borders the other variant does not. The rules that would be required to create an algorithm would be quite complex.

Vitiligo Timeline and Epidemiology

The timeline of vitiligo can vary widely from person to person. In general, the onset of vitiligo is gradual, with the appearance of white patches on the skin that gradually enlarge and spread over time. Here is a rough timeline of the progression of vitiligo:

1. First signs: In the early stages of vitiligo, a person may notice small, white patches of skin on their body. These patches may be round or oval in shape and have a well-defined border. They may also be symmetrical, meaning they occur in the same location on both sides of the body.
2. Spreading: Over time, the white patches may enlarge and spread to other areas of the body. The rate of spreading can vary widely from person to person and may be slow or rapid.
3. Fusion: In some cases, adjacent white patches may merge, resulting in larger areas of depigmentation.
4. Stability: In some cases, the spread of vitiligo may stop on its own, resulting in a stable pattern of depigmentation. This is more likely to occur in people with localized vitiligo, which affects only a small area of the body.
5. Regression: In rare cases, the depigmented areas may begin to pigment on their own, resulting in a gradual return of normal skin color.

It is important to note that the course of vitiligo can be unpredictable, and the timeline of the disease can vary widely from person to person. In some cases, vitiligo may progress rapidly and result in extensive areas of depigmentation, while in other cases it may progress slowly or remain stable for many years.

Vitiligo is a relatively common skin disorder, affecting approximately 1-2% of the global population. The condition can affect individuals of all ethnicities and skin types, although it is more common in people with darker skin tones. Here are some key epidemiological factors of vitiligo:

1. Age of onset: Vitiligo can occur at any age, but most commonly begins in individuals between the ages of 10 and 30.
2. Gender: Vitiligo affects both males and females equally.

3. Family history: A family history of vitiligo is present in approximately 20% of affected individuals, suggesting a genetic component to the condition.
4. Autoimmune disorders: There is a strong association between vitiligo and autoimmune disorders, such as thyroid disease and type 1 diabetes.
5. Environmental factors: Exposure to certain environmental factors, such as chemicals or physical trauma, may trigger the onset of vitiligo in some individuals.
6. Disease course: The course of vitiligo can vary widely from person to person, with some individuals experiencing rapid progression of the disease, while others have a stable or slowly progressive course.

Overall, vitiligo is a complex disorder with a range of potential risk factors and disease courses. While there is no cure for the condition, a variety of treatment options are available to help manage symptoms and improve the quality of life for affected individuals [7| [16].

Prevalence between 0.14 and 8.8% are reported from different races [1][3]. While vitiligo occurs in northern Europe in 0.5% of cases [16], it becomes observed in 1% in the USA [17]. In areas with strong sunshine irradiation and darker skin types, such as in patients in India and the Far Eastern countries, vitiligo seems to be much more common [1, 3]. Although some studies showed a clear preponderance of females was observed in some patients, most authors now assume that both sexes are affected equally often. This discrepancy, which was also noticed in other studies, with a heightened women's attention and concern about one in the first-place cosmetic defect [3].

Vitiligo is an acquired, idiopathic disease [7]. She can occur with age normal and will manifest from a few days to weeks after birth [8][18] up to a first development at the age of 81 years reported [3]. In 50% of cases, the disease manifests before the age of 20 years [17] and in 70-80% between the ages of 10 and 30 years [8].
40.9% (109 of 264) of patients reported the occurrence of Vitiligo before the age of 20.

According to the age of manifestation, SCHALLREUTER divided vitiligo into three categories and those were divided further into classes [7]:

- Children's vitiligo: manifestation age 1.-12. age
- Adolescent vitiligo: manifestation age 13.-20. age
- Adult vitiligo: age of onset 21 years

These groups sometimes show significant differences in the prognosis as well as the familial accumulation of the disease, the occurrence of a Koebner phenomenon, and the emergence of special forms, such as premature graying (canities praecox) or

halo naevi [7][19]. The Koebner phenomenon, named after Heinrich Koebner, is a characteristic skin response in which new skin lesions appear at the site of trauma or injury on previously unaffected skin. It is commonly observed in various skin conditions, including vitiligo

Etheology, Histopathology, and Patient Material in Vitiligo Research

Views on the inheritance of vitiligo remain controversial to this day. A familial accumulation of Vitiligo is not found regularly. However, familial significantly increased accumulation with 46% in the subgroup of children's vitiligo (age of manifestation 1 to 12 years of age) [7]. These observations support the hypothesis of the involvement of hereditary factors in the development of vitiligo[3].

The typical histopathological changes in the skin of vitiligo patients affect not only the melanocytes but also the keratinocytes. Keratinocytes produce keratin, a protein that provides structural support to the skin and helps form its barrier function. In the epidermal unit, each melanocyte supplies approximately 36 keratinocytes via its dendrites, transferring mature melanosomes containing pigment to them.[7].

With vitiligo it is the absence of melanin in keratinocytes characteristic, its occurrence of degenerated melanocytes without melanin formation, and a reduced number of pigment-forming cells [6][8][9].

The dendritic processes of the Melanocytes decrease in number and an intercellular one develops a Gap between melanocytes and keratinocytes [29].

Pathological changes are also found in melanocytes at the periphery of the vitiligo focus. These changes include fatty degeneration, pyknosis, cytoplasmic vacuolation, and aggregated melanosomes.[30].

The image material contained the complete set of 69 patients. Age, skin type, and vitiligo severity were a mix that was important to produce good ground truth data. The patient data contains images of complete body images (all four sides) and some additional positions. The main work was targeted at the patient's head. The head and hands are the most prominent parts not easy to cover in public. The images are also taken in a time series with an interval of 6 months. The patient treatment results can therefore be proven. The current clinic requires a human calculation on the visual and calculates the vitiligo percentage by use of the hand. Data was obtained under ethical considerations, informed consent was obtained from all participants, including patients whose medical images were utilized in the study.

The goal is to develop a measurement algorithm that collects data and enables more robust and comparable results. Furthermore, the scope of the work was extended to analyze complete body images, including the chest, legs, arms, and feet.

The dataset used for this study consists of 69 images of patients, captured at different time points. The complete photo set for each patient comprises 18 images, including frontal, side, and back views. Each side contains four images dividing the patient into head, top, middle, and lower sections, allowing for a comprehensive examination of the affected areas.

A retrospective, non-interventional, monocentric center study was performed in the Department of Dermatology of the University Hospital of Bordeaux. This study was approved by the local committee of the University Hospital of Bordeaux. All patients gave their written and oral consent.

Clinical pictures of the face were taken using the FOTOFINDER® bodyscan system combined with two high-output UV flashes. The photographs were standardized in a neutral position.

One hundred vitiligo pictures were selected, 80% of which were used for the training and the remaining 20% for validation of the trained model. As a final test data set, 69 additional new pictures were used. These images corresponded to 59 adult patients, of which 10 were reevaluated 6 months after an intervention therapy. Three expert physicians estimated the proportion of vitiligo lesions at two time points. A global evaluation was performed from 0% (no vitiligo lesion) to 100% (complete depigmentation). Inter- and intraraters reliability were evaluated by comparing the scores between experts at each evaluation, and from two separate time points (test-retest) respectively. A two-way random model was performed for intraraters (and a two-way mixed for intraraters). Absolute agreement and single-measures intraclass correlation coefficient (ICC) were also estimated. Inter-rater reliability was evaluated between experts and AI. For non-normally distributed data, a logarithmic transformation was performed. The following guidelines for the interpretation of the ICC were used: < 0.4 was considered poor, 0.4–0.59 fair, 0.6–0.74 good, and ≥0.75 excellent (Cicchetti 1994).

Female patients represent 69.5% with a mean age of 38.7 years(±14.05 years). The mean duration of the disease was 14.2 years (±14.53 years). The majority of patients were skin phototype I-II-III (76.3%). All patients were mainly followed for non-segmental vitiligo (acrofacial 6.8%, generalized 91.5%), except the patients who presented segmental vitiligo.

The images were acquired in the form of graphic images, utilizing color schemes. Considering the human visual system's perception and the role of color in image composition, the blue color channel emerged as the most informative component. Most computer graphic algorithms are designed to work on single-channel or grayscale images, necessitating the extraction of relevant information from the blue channel.

To address these challenges, a range of computer graphics methods were employed in attempts to overcome the aforementioned obstacles. Some of these methods yielded promising results, while others proved ineffective in solving the unique complexities associated with vitiligo analysis.

Computer Graphics in Vitiligo Research

To start using AI (Artificial Intelligence) or Machine learning we need to apply computer graphic methods on the images to extract the object of interest. In image processing, various techniques are employed to extract valuable information from images. Mostly a combination of multiple techniques is necessary. Computer graphics has been a well-established field since the early days of computer science, focusing on image processing, programming languages, and algorithms to enhance object detection, background detection, and face recognition. In the context of vitiligo localization, these methods play a crucial role and will be discussed in detail in the following sections.

Computer Graphic methods are required to analyze image data. Based on the requirement to locate vitiligo we are discussing a set of methods that were tested to be used for vitiligo detection. The methods are general Computer Graphic methods used on images of vitiligo patients. One basic question is the preferred programming language to use.

Programming Languages

The selection of a programming language for a specific project or need is sometimes quite mandatory. The available interfaces or libraries that can provide necessary features might be important. Several programming languages are commonly used for AI programming, including:

1. **Python**: Python is currently one of the most popular programming languages for AI development due to its simplicity, ease of use, and large number of available libraries and frameworks such as TensorFlow, PyTorch, and Scikit-learn.

2. **C++**: C++ is a high-performance language that is often used for developing AI and machine learning applications that require efficient processing and high-speed performance.

3. **R**: R is a language commonly used for statistical computing and data analysis, making it popular for machine learning and AI applications.

4. **Java**: Java is a general-purpose programming language that is widely used for enterprise applications, including AI and machine learning.

5. **Julia**: Julia is a relatively new language that was designed specifically for scientific computing and numerical analysis, making it a good choice for AI and machine learning applications.

Ultimately, the choice of the programming language will depend on the specific requirements of the project and the preferences of the developer there is not per se the best programming language, each programming language has its advantages but also some limitations.

Python as a Programming language

Python is a high-level, interpreted programming language that is designed to be easy to read and write. It was first released in 1991 by Guido van Rossum and has since become one of the most popular programming languages in use today.
Python is known for its simplicity, flexibility, and readability, which makes it an excellent choice for beginners and experts alike. It has a vast and growing number of libraries and frameworks that can be used for web development, data analysis, scientific computing, machine learning, and artificial intelligence.

Python is also cross-platform, which means it can run on various operating systems, including Windows, macOS, and Linux, and it is an open-source language, which means it is free to use and can be modified and distributed by anyone.
Python code is typically written in plain text files with a .py file extension and can be executed by running the Python interpreter on the command line or through an integrated development environment (IDE) such as PyCharm, VS Code, or Spyder.
The Libraries required can be easily downloaded by use of Git resources or packaging information. This makes it quite easy to follow the new releases. Nevertheless, major changes as we can see in the area of Neural Networks developments such as Tensorflow or Keras undergo a quite radical code change which requires not only the update of existing libraries to a new version but also major code changes due to changing APIs.

The alternative would have been to use C or C++ which is more complex and not that easily expendable for other libraries. The programs need to be compiled for

each operating system. That makes interoperability more difficult. Since the introduction of Java and the byte code that can be executed on almost all hardware that offers an interpreter the usage of Java programs exploded. Today Java is almost used everywhere even in washing machines.

The modularity and extensibility of Python make it a perfect programming language for Computer Vision and also the AI. Most of the AI and C++ libraries are available for Python and can be used.

Based on the comparison of C/C++ and Python we decided to use the Python language for image analysis. This leads to the next important point the source to be analysed the patient images.

C or C++ as a Programming language

C and C++ are two closely related programming languages that are commonly used for developing system software, operating systems, and embedded systems.

C is a general-purpose, procedural programming language that was developed in the early 1970s by Dennis Ritchie at Bell Labs. It is a low-level language that provides direct access to memory and hardware, making it a popular choice for developing operating systems and embedded systems that require efficient use of resources. C is also widely used for writing applications in various domains, such as scientific computing, gaming, and finance.

C++ is an object-oriented programming language that was derived from C in the 1980s by Bjarne Stroustrup. It is designed to provide features such as encapsulation, inheritance, and polymorphism, which allow developers to write complex software systems in a modular and extensible manner. C++ is used in a wide range of applications, including operating systems, video games, web browsers, and scientific computing.

Both C and C++ are powerful languages that offer high performance, low-level access to system resources, and efficient memory management. However, they also require more complex programming techniques compared to higher-level languages such as Python or Java.
R, Java, and Julia were not relevant since those are less integrated with computer graphic libraries. We therefore were choosing Python as a programming language for this research.

Data as foundation

In the field of AI, data serves as the fundamental building block for analysis and training. For this research, the data utilized was obtained from the clinic research

conducted at the University of Bordeaux over several years. It is widely acknowledged that the quality and quantity of data play crucial roles in the effectiveness of AI systems. While the quantity of data required may vary depending on the Neural Network architecture employed, it is the quality of the data that ultimately determines the success of training and validation processes.

To ensure optimal training of the AI models, the data used in this study adhered to specific quality measures. Images were carefully selected, with strict criteria in place to eliminate instances of overexposure or blurriness. By curating a high-quality dataset, we aimed to enhance the accuracy and reliability of the trained neural networks.

It is important to note that data is not only essential for training AI models but also for evaluating their performance. Validation and test data enable researchers to assess the generalisability and robustness of the developed models. By incorporating well-curated data into the research process, we can establish a solid foundation for the AI system, enabling more accurate and reliable results.

Computer Vision and AI in Vitiligo Research

Computer vision and AI (Artificial Intelligence) are two related fields that work together to create intelligent systems that can understand and interpret visual data. Computer vision is a field of study focused on teaching computers to interpret and understand visual information from the world around them, such as images and videos. It involves the use of algorithms and mathematical models to analyze and interpret visual data and has applications in a wide range of fields, including healthcare, entertainment, security, and transportation.

AI, on the other hand, is a broader field that encompasses all aspects of creating intelligent systems that can perform tasks that normally require human-level intelligence, such as learning, reasoning, and problem-solving. AI systems can be designed to work with many different types of data, including text, speech, and images, and often use techniques such as machine learning, deep learning, and natural language processing to analyze and interpret data.

In recent years, computer vision and AI have become increasingly integrated, with advances in machine learning and deep learning techniques allowing computers to more accurately and efficiently analyze and interpret visual data. This has led to the development of intelligent systems that can perform a wide range of tasks, such as object recognition, face recognition, autonomous vehicles, and medical image analysis.

Object recognition is the field of computer vision that helps computers identify and understand objects in an image. Within the area of object recognition, there will be different approaches, including deep learning and convolutional neural networks (CNN), which are often associated with artificial intelligence. CNNs are responsible for helping computers recognize images at a detailed level.

CNNs also help provide facial recognition in photos, distinguishing between the eyes, mouth, nose, and other features of a person's face. This information can be used for several different purposes in the real world, such as identifying people and fetching relevant information about them. In general, the CNN requires a lot of data to be trained and to provide acceptable accuracy in the predictions that are taken. This leads to the problems of data generation training and validation of the CNN Models.

In general, it can be said that Computer Vision is the ground for AI models to be created. Without basic Computer Vision, there is no AI possible. The first step is to locate the objects of interest this is archived with background detection and discussed in the next section.

Image Processing in Vitiligo Research

Image processing is a fundamental technique utilized for manipulating and analyzing digital images. It encompasses a broad range of methods aimed at enhancing and improving the quality of digital images. These methods find applications in various fields, including computer vision, medical imaging, robotics, and remote sensing. Image processing involves the use of algorithms to perform operations such as filtering, transformation, segmentation, feature extraction, and pattern recognition on digital images.

In the realm of image processing, several facets are crucial to consider, particularly when it comes to image analysis. One such facet is background detection, which involves locating the object of interest within an image. For the human visual system, aided by accumulated knowledge and memory, this task is typically performed instantaneously. However, for a computer, the situation is different. The digital representation of an image, stored as a sequence of zeros and ones, lacks inherent meaning. It is through the identification of patterns or the grouping of information within these binary sequences that objects can be detected. However, the challenge lies in determining which objects we seek to identify.

In the absence of contextual cues, image analysis becomes arduous. The vastness of the universe renders it impossible for humans, or computers, to recognize every object without a relevant context. Even for humans, detecting objects within a scene

can be challenging if there are no hints or clues provided as to what to look for. This phenomenon becomes particularly evident in images where objects are concealed or camouflaged. Initially, the human eye may not perceive all objects present, but upon receiving a prompt or cue, attention is directed toward searching for specific shapes or characteristics. For a computer, these shapes or characteristics are the patterns it must seek. From a human perspective, comprehending the complete image relies on separating the background from the foreground, with the foreground containing the information of interest. However, when encountering images that lack clarity or an explicit context, such as those created by Escher, even the human visual system, equipped with memory (though not always trained or attuned in a specific manner), can be deceived by these optical illusions.

Moving beyond mere object detection, a critical aspect of image processing is object classification. This process entails examining the identified objects using specific defining attributes to categorize them accurately. Classification poses a challenging task, and multiple possibilities or combinations thereof can be employed. The following section highlights some of the key methods from the computer graphics field to utilize for classification.

Background Detection for Vitiligo Research

Background detection plays a crucial role in image processing, as it involves distinguishing between the background and the foreground objects within a scene. This task is challenging, as determining what constitutes the background and the objects of interest often requires careful examination and analysis. Even for humans, understanding an image may require multiple observations to differentiate between the background and the foreground information.

Recent advancements in research, leveraging large datasets such as the one provided by Facebook, have shown promising results in object detection based on selected images. These approaches have demonstrated high accuracy in detecting objects within images that contain recognizable patterns. However, their effectiveness in detecting non-pigmented skin or addressing specific challenges related to vitiligo detection is limited.

The challenge of background detection can be viewed as a problem of defining the boundary between relevant and irrelevant data. The variability in images and the objects to be detected further complicates this task. Humans utilize their cognitive abilities and prior experience to assist in locating objects and separating the foreground from the background. This initial step of distinguishing between the two layers is the foundation for further analysis and processing.

Exploring different options for background detection and identifying the best use cases is essential. When no prior information about the objects in the image is available, the task becomes particularly difficult. In such cases, one approach is to identify moving objects, as their motion can provide clues about their boundaries and aid in background detection.

Background detection is a fundamental technique in image processing, enabling numerous applications such as video surveillance, object detection and tracking, and motion analysis. Additionally, it can be employed to enhance image quality by eliminating noise and undesired elements from the background. A range of algorithms and methods exist for background detection, encompassing simple techniques like thresholding and frame differencing to more sophisticated approaches involving statistical modeling and machine learning. The selection of the most appropriate method depends on the specific application requirements and the characteristics of the image data being analyzed.

We were first focusing on the head. The AI models for finding a head in an image worked and we were able to create a bounding box around the area and crop it for processing. This worked quite efficiently. When we look at the other images containing a person in different positions its complexity is growing. The question of where and what the object is at first opens. We can not tell what the scene is containing. Finding the background in an image is not an easy task as it seems to be at first sight. The information in the image is for first look all the same as the knowledge a human has gathered of the time the cognitive intelligence is building objects and therefore finding the object automatically separating the background. For a computer, this task is due to the lack of memory or intelligence not that easy to perform.

Skin Detection in Vitiligo Research

Skin detection in images is identifying and separating the pixels or regions in an image that correspond to human skin. It is a fundamental step in many applications of computer vision, including face detection, gesture recognition, and image analysis.

The process of skin detection typically involves analyzing the color and texture properties of the image to distinguish between skin pixels and non-skin pixels. Skin pixels are generally characterized by their color, which falls within a specific range of hue, saturation, and brightness values. Additionally, skin pixels tend to have a relatively uniform texture and intensity, which can be used to further distinguish them from the background.

Skin detection algorithms may use a variety of techniques to identify skin pixels in an image, including color-based segmentation, texture analysis, and machine-learning

approaches. The accuracy of skin detection depends on various factors, such as lighting conditions, skin tone variations, and the presence of non-skin colors in the image.
Skin detection has many practical applications, such as in the development of facial recognition systems, tracking human movements in video surveillance, and medical imaging for the diagnosis of skin diseases.

If we add information about the object that we are looking for we can introduce a different technique to locate the skin in an image. Human skin contains a variety of different skin colors we see very light colors in Japan to very dark colors in Africa. This variety is difficult to capture. Different image formats offer different advantages.

BGR is a color model used in digital image processing, where it represents the color of each pixel as a combination of blue, green, and red components in that order. The BGR color model is commonly used in computer vision applications and is the default color model used by OpenCV, a popular computer vision library.

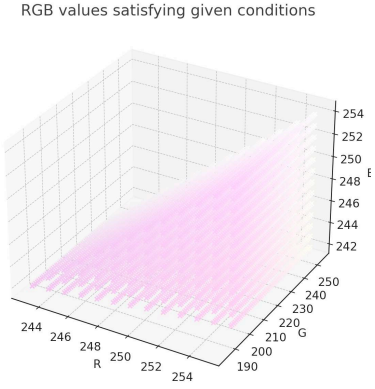
In the BGR color model, each pixel is represented by three 8-bit values, where the first value represents the blue color component, the second represents the green component, and the third represents the red component. This order of components is different from the more commonly used RGB color model, where the order of components is red, green, and blue.

The BGR color model is used by OpenCV and some other computer vision libraries because it is optimized for efficient processing of image data in certain operations such as color space conversions and matrix operations. However, it is important to note that when working with images in other applications, it may be necessary to convert between the BGR and RGB color models to ensure the correct interpretation of the image data.

The image information is stored in three layers which are interpreted as Blue, Green, and Red. The channels contain the color information that a pixel contains. All pixels contain a specific amount of each channel and therefore display a specific color. The combination of the three colors makes it possible to create almost every color possible.

For the detection of human skin, the values fall in a certain combination of the three values and can be described as follows:

$$(R > 205), (G > 151), (B > 241),$$
$$(\max\{R, G, B\} - \min\{R, G, B\}) < 60,$$



$$|R - G| < 60,$$
$$R > G,$$
$$R > B$$

The above conditions need to be true then the color displayed is considered to be skin tone.

The calculation is quite complex which indicates that the BGR format is not suitable for skin detection. There is not a real cluster for the skin values which makes it hard to define a range.

YCrCb is a color model used in digital image processing, where it represents the color of each pixel as a combination of luminance (Y), chrominance red difference (Cr), and chrominance blue difference (Cb) components. The YCrCb color model is commonly used in video and image compression, digital broadcasting, and image processing applications.

In the YCrCb color model, the Y component represents the brightness of the pixel, while the Cr and Cb components represent the color difference between the pixel and its corresponding white point. The YCrCb color model is designed to be more perceptually uniform than other color models like RGB, making it more suitable for the compression and processing of image and video data.

The YCrCb color model can be represented in different bit depths, with 8-bit and 16-bit representations being the most common. In an 8-bit YCrCb image, each pixel is represented by three 8-bit values, where the Y component ranges from 0 to 255, and the Cr and Cb components range from -128 to 127.

In image processing, the YCrCb color model is often used for operations such as image compression, color space conversions, and color correction. It is also commonly used in video processing applications, where it provides a more efficient representation of the color information in a video stream.

If we look at BGR images the information of one color channel is due to the presence of the base colors in a color not distinct. Each image contains the three colors as a mix and therefore an overlap of all three Channels creates the color information. That means that there is redundant information in case we can describe the color differently. For TV broadcasts the transmission time is vital here research was conducted and a new image format was created that provides a more efficient storage for the image information. The redundancy is reduced.

YCbCr is an approximation to color processing and perceptual uniformity, where the primary colors corresponding roughly to red, green, and blue are processed into perceptually meaningful information. The transformation of a BGR to YCrCb is therefore not a very cost-intensive calculation. By doing this, subsequent image/ video processing, transmission, and storage can do operations and introduce errors in meaningful ways. YCbCr is used to separate a luma signal (Y) that can be stored with high resolution or transmitted at high bandwidth, and two chroma components (CB and CR) that can be bandwidth-reduced, subsampled, compressed, or otherwise treated separately for improved system efficiency.

One practical example would be decreasing the bandwidth or resolution allocated to "color" compared to "black and white" since humans are more sensitive to black-and-white information.

The benefits of image transmission do not lead to the same advantages that lead to skin detection. The definition of a skin color region is quite complex and therefore not very efficient. To locate a cluster of skin collars in the YCrCb definitions this would require 5 equations to define a section in the space. If we try to use it on the Y channel we will have almost the same information as in the blue channel. The information in Y is the light in formation. The skin or sections that are less colored are lighter therefore the Y information is higher. We can define the following range to acquire the biggest range of skin tones:

```
# Skin area
Minimum [40,113,90]
Maximum [255,180,190]
```

For YCrCb this range is creating a skin area that matches most of the images provided. The range would need to be adapted in case skin colors not yet processed would be introduced.

HSV (for hue, saturation, value) is an alternative representation of the BGR color model, designed in the 1970s by computer graphics researchers. Its benefits lay in the perception of the human eye to more closely align the colors and the color-making attributes. In these models, colors of each hue are arranged in a radial slice, around a central axis of neutral colors which ranges from black at the bottom to white at the top. The HSV representation models how colors appear under light.

The advantage of this color model is due to the close connection of the same colours the grouping allows it to locate clusters of colors and to group them. The definition of H value defines the color and the S and V channels the intensity and amount of light.

That means with the same H and S values when changing the V Chanel we can keep the same color. This has enormous benefits when looking at images where the lightning is generating shadows. The shadow areas have the same color but less light therefore changing the V in the image will keep the color unchanged.

The HSV model proved to be the most suitable model for the skin detection. The skin tones can be located in a certain range for a lot of skin temperatures.
The following values describe the skin range.

(hMin = 4 , sMin = 67, vMin = 31),
(hMax = 179 , sMax = 255, vMax = 178)

The calculations to convert a BRG to an HSV image are expensive and can be calculated quite easily. The best approach is to combine all three skin masks and use the resulting mask for skin detection.

TIFF (Tagged Image File Format) is a popular file format used for storing digital images, especially in the fields of photography, printing, and publishing. It was developed by Aldus Corporation in collaboration with Microsoft in 1986 and is now owned by Adobe Systems.

TIFF is a versatile format that supports a wide range of color depths, resolutions, and compression methods. It is also capable of storing multiple images within a single file, allowing for easy organization and management of image data.

One of the key advantages of the TIFF format is its ability to store images in a lossless format, meaning that no image data is lost during compression. This makes TIFF a popular choice for archiving and preserving high-quality images, especially in the printing industry.

TIFF files can also be compressed using a variety of methods, including LZW, ZIP, and JPEG, which allows for a significant reduction in file size while maintaining high image quality. However, it is important to note that some compression methods may result in loss of image data, and therefore it is important to choose an appropriate compression method based on the specific requirements of the application.

TIFF is supported by most image editing and processing software, making it a widely used format in the industry. However, its relatively large file size and lack of support for some advanced features like transparency and animation make it less suitable for certain applications.

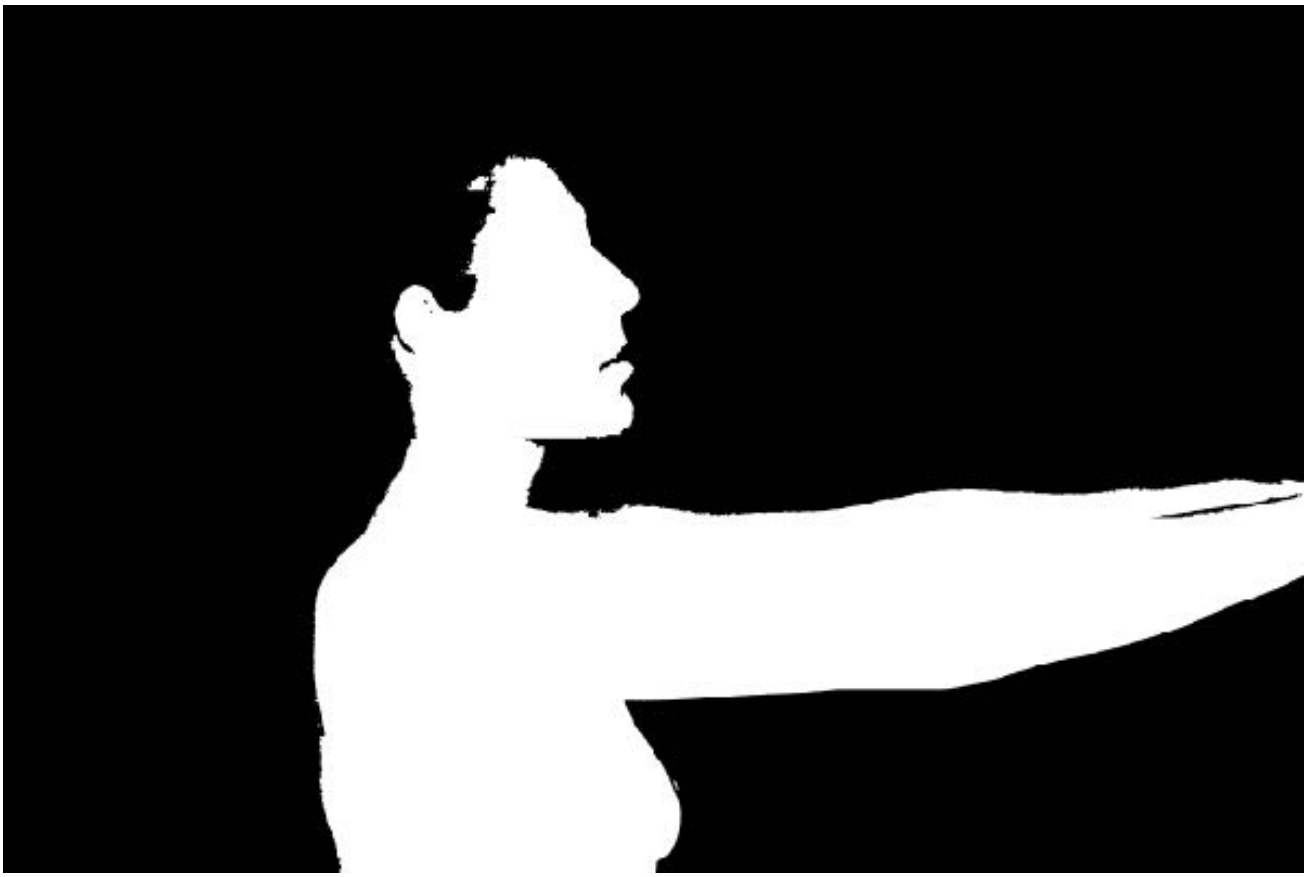
It is worth noting that TIFF files saved in a lossless format may result in larger file sizes compared to other image formats that use lossy compression methods, such

as JPEG. However, lossless compression ensures that no image data is lost, making TIFF files in lossless format a popular choice for archiving and preserving high-quality images.

It is not used in the image processing or skin detection. This format is used in AI training and model creation. This format is most suitable for image training.

If we apply the above setting to the image and create a mask for each of them which will then be combined into one single mask we can generate a mask that contains almost all skin shown by the object in question. The generation of masks with

Figure 3) Object Detection Combination of Masks



different color schemas allow us to capture all different skin tones. The skin that is hidden by clothing or by hair is not taken into account and can not be added a sample is shown in Figure 3) where a complete mask is generated.

Skin Color in Vitiligo Research

If we narrow down our focus to detecting human skin, we can employ a different approach for background separation. The use of a color range to define human skin can help locate relevant objects in an image. However, this method is only effective when the background does not contain colors similar to human skin, as it would fail in scenarios where, for example, the background consists of sand or earth tones closely resembling skin color. With a combination of all color scheme masks, we can create a mask that is quite solid for all combinations of skin colors.

HEX color scheme vs HSV color schema

Both HEX and HSV are color representation systems, but they serve different purposes. HEX is primarily used in web design and digital graphics to specify colors on computer screens, while HSV (Hue, Saturation, Value) is a color model often used in image processing and computer vision applications to describe color properties.

Object Detection in Vitiligo Research

Object Detection is a crucial field within the domain of Computer Vision, focused on the task of accurately locating and identifying specific objects or regions of interest within an image. The ability to effectively detect objects plays a pivotal role in various applications, ranging from autonomous driving and surveillance systems to medical imaging and augmented reality. By enabling machines to perceive and understand their visual environment, object detection empowers AI systems to make informed decisions and take appropriate actions based on the detected objects.

In the realm of AI research, object detection has garnered significant attention due to its paramount importance in numerous real-world scenarios. Detecting objects accurately and efficiently serves as a fundamental building block for higher-level tasks such as object recognition, tracking, and scene understanding. Moreover, object detection lays the foundation for advanced applications like semantic segmentation and instance segmentation, enabling a more detailed and comprehensive understanding of images.

The field of object detection encompasses a diverse range of techniques and methodologies, each with its strengths, limitations, and trade-offs. Traditional approaches, such as the popular Viola-Jones algorithm [65], employ handcrafted features and classifiers to detect objects based on predefined patterns or characteristics. More recently, deep learning-based approaches, particularly those utilizing Convolutional Neural Networks (CNNs), have revolutionized object detection by leveraging the power of deep feature representation learning. Techniques like region-based methods (e.g., R-CNN, Fast R-CNN, Faster R-CNN) and single-shot

methods (e.g., YOLO, SSD) have achieved remarkable performance gains in terms of both accuracy and speed.

While object detection has opened up new possibilities and advancements in AI, it also presents certain challenges and considerations. One critical aspect is the need for large and diverse annotated datasets to train robust and generalizable object detection models.

Collecting and annotating such datasets can be time-consuming and resource-intensive, requiring significant human effort and expertise. Additionally, striking a balance between detection accuracy and computational efficiency is an ongoing challenge, especially in real-time applications where processing speed is crucial.

Furthermore, object detection algorithms may exhibit biases or limitations when faced with complex or novel scenarios. Factors such as occlusion, scale variation, background clutter, and viewpoint changes can pose difficulties in accurately detecting objects. Adapting object detection models to handle these challenges and ensuring their robustness in diverse real-world environments remains an active area of research.

In this research, we aim to explore different possibilities and methodologies for object detection, delving into the strengths, limitations, and potential implications of each approach. By examining the usage, positive impacts, and potential drawbacks, we seek to contribute to the advancement of object detection techniques, further enhancing their effectiveness and applicability in AI systems.

Moving Object Detection

One approach to detecting moving objects is by utilizing multiple frames of data, similar to how it is done in movies. By comparing consecutive frames, we can identify the presence of moving objects and filter them out from the static background. This technique is commonly used in video surveillance and motion detection systems.

To apply a similar concept to still images, we can emulate the movement of objects by performing a series of resizing and shift operations. This process involves systematically transforming the image at different scales and positions to create a sequence of images. By comparing these transformed images, we can extract the silhouette or outline of the moving object.

The resulting silhouette provides valuable information about the shape and position of the moving object in the image. This technique is particularly useful when dealing

with dynamic scenes or scenarios where object movement is a critical factor to consider. It enables us to separate the moving objects from the background and focus on analyzing their characteristics and behavior.

By incorporating this approach into image processing algorithms, we can enhance the detection and analysis of moving objects in various applications, including surveillance, autonomous vehicles, and action recognition. The utilization of multiple frames and the generation of object silhouettes facilitate a more comprehensive understanding of the dynamics within an image and enable accurate tracking and identification of moving objects.

In summary, the use of multiple frames and silhouette generation techniques enhances the detection and analysis of moving objects in both video and still images. This approach expands the scope of object detection beyond static scenes, enabling the identification and tracking of objects based on their dynamic properties.

Adapted Moving Object Detection

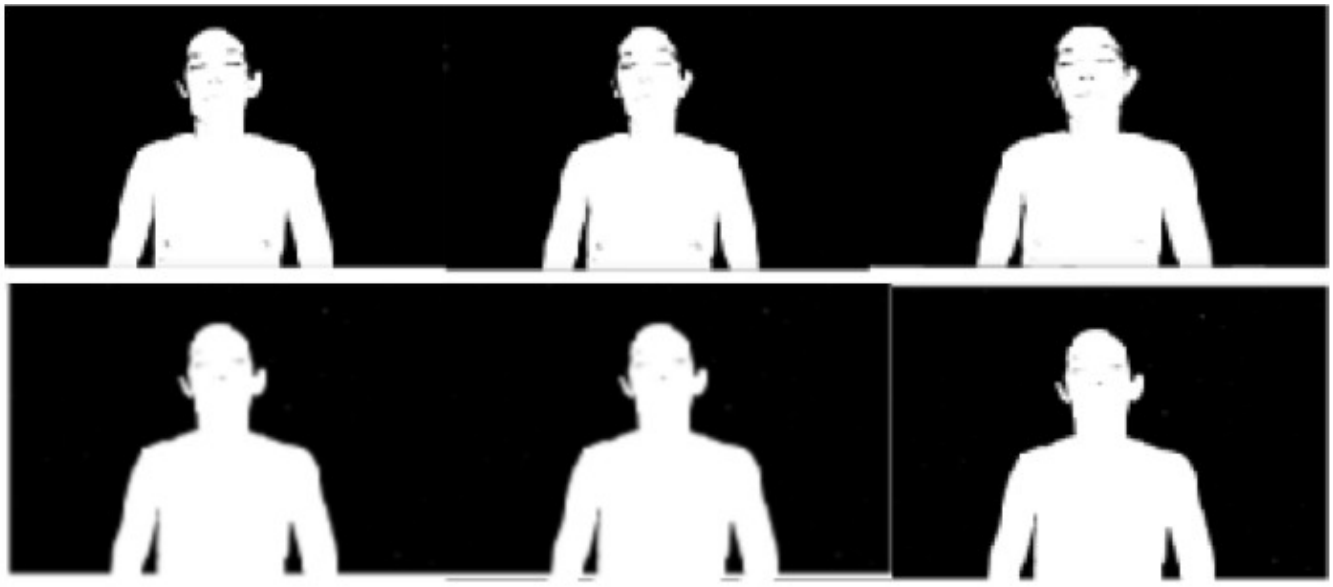
In the context of video processing, a common approach to distinguish between foreground and background is by comparing a sequence of images over time. Moving objects are considered as the foreground, while unchanged pixels represent the background. This method assumes a stationary camera capturing the same scene over a period.

However, the problem becomes more challenging when dealing with moving cameras. In such cases, alternative techniques need to be employed. One of these techniques is the use of the BackgroundSubtractorKNN class, which utilizes a background image to separate the foreground. Multiple images, such as a video sequence, are analyzed to identify stable points. This approach becomes even more complex in outdoor scenes due to the changing lighting conditions caused by variations in sunlight. The background should be very stable. If the background as reference is also moving the noise created is disturbing the object detection. Small changes can be filtered out but larger sections get problematic.

To adapt this technique for single images, an innovative strategy was devised. A background image was created by removing the patient and replacing the corresponding area with the surrounding background in our case the black fabric is used as a reference. This background image served as a baseline. The background looks only solid in the image if we look at the grey image or just one channel we can see that we have fine wooden fabric and not a solid black block. The patient image was then resized by a few pixels and shifted horizontally, creating a simulated movement. In Figure 4) the process is sketched. We start from left to right the image is moved down for 2 pixels. The next step is a resize of 2 pixels and again a moment

to the right was carried out. This process was repeated several times to obtain an accurate result. Each movement optimizes the mask and reducing more and more background.

Figure 4) Adapted Background Detection



Simulated Movement of image from left to right

Face Recognition

How to locate a face in an image? This is a difficult task to analyze an image for a special object of interest. To find a face it's best to locate first an object that contains a face to not locate a false object as a face. Face recognition is a sophisticated process used to identify or verify an individual's identity by analyzing their facial patterns, features, and characteristics. It relies on computer vision and machine learning algorithms to extract and analyze facial attributes, including the distance between the eyes, the shape of the nose, and the overall facial contour. This technology finds applications in various domains, such as security systems, access control, surveillance, and social media.

In our research, we utilized face recognition techniques to locate the faces of patients with vitiligo. Initially, we experimented with the Idlib Library, an image library

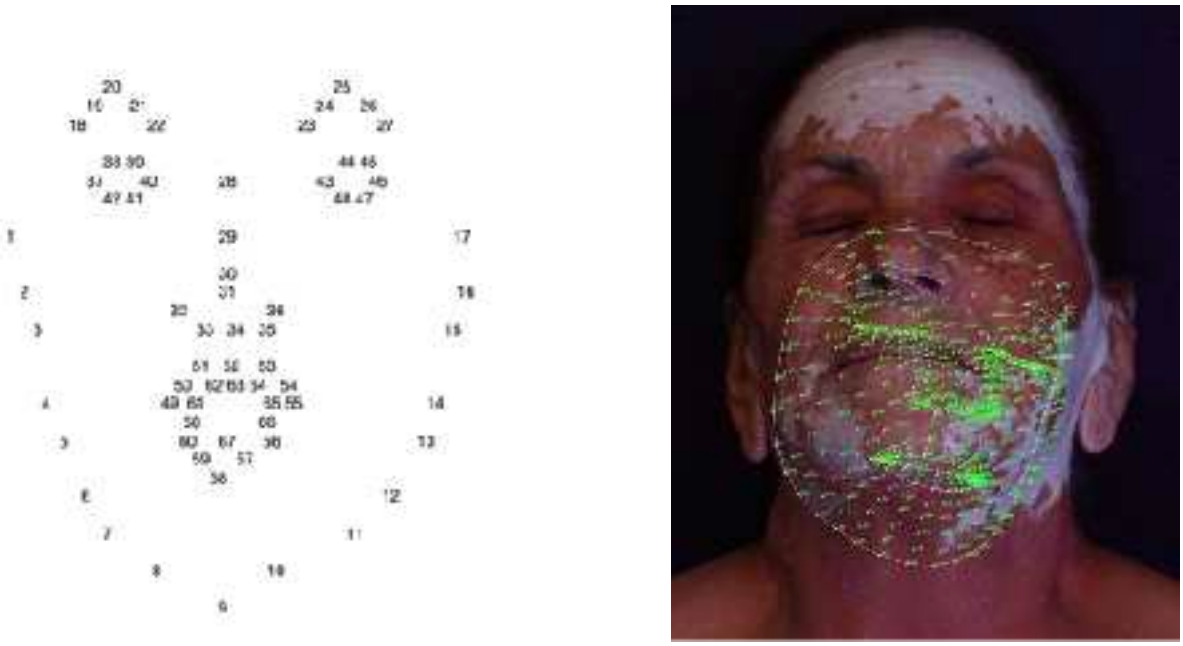
commonly used for normal person face recognition. However, we encountered challenges specific to vitiligo cases, as the condition results in different shading patterns compared to regular faces. For instance, dark areas around the eyes, which are typically associated with facial features, appeared lighter due to vitiligo patches. Additionally, vitiligo patterns on the chest were mistakenly identified as faces, leading to inaccurate processing.

To address these issues, we turned to the Google Library for Face Recognition, which yielded significantly better results. The face detection accuracy improved to 95%, indicating its effectiveness in identifying the presence of a face. However, capturing side views of the face remained challenging due to the complex nature of such images, compounded by the presence of vitiligo patches. To mitigate this limitation, we adopted an alternative approach: selecting a fixed area in the image as a rough estimation of the face's location. While this technique helped reduce false-positive detections, additional features, such as the location of the nose, ears, and eyes, became crucial as further requirements emerged in our research.

Face Detection

Face detection is a more detailed recognition it is not only focused on the presence it is also focused on the positions of the parts like eyes, nose, mouth, and other features Figure 5A).

Figure 5) Facial Points and Face Detection



A) Facial Points

B) Facial Points on Vitiligo Faces

Especially in the context of vitiligo patients, poses unique challenges for AI models that are primarily trained on images of individuals with pigmented skin. The distinct skin patterns present in the faces of vitiligo patients do not align well with the patterns recognized by these models. This discrepancy becomes evident when observing the images below Figure 5B), where certain parts of the face are inaccurately detected. Not only that other areas of the body can be recognised as a face in case some parameters trigger the network to match certain high and low value points. The current networks use more training data and more reference or key points for adequate evaluation of the images. But with the nature of vitiligo that normal dark parts are now light, the networks can not function accurately.

Using large bounding boxes around the located face points we achieve a best guess. We might get some of the background but we will capture the whole head. The networks are trained mostly for portrait images while the side views are less recognised.

Face Overlay

Since we got two images of the patient after a certain time the idea arose to make an overlay of the two images. This required the location of the face discussed in the previous section. After successfully detecting the face in an image, a compelling idea emerged: utilizing face overlay to visualize the differences in vitiligo treatment over a time series. By comparing the first visit image with subsequent visits, we can effectively illustrate the changes in the structure of vitiligo.

Face overlay involves analyzing the facial features and identifying unique landmark points that can be found in both images. Matching these landmark points entails a series of transpose and move operations, aligning the second image with its corresponding points in the first image. The closest matching points are connected to create a visual representation of the transition.

To achieve this, a transition matrix can be calculated based on the location of the face points. It is important to have a minimum of four accurately located points for the transformation. The distribution of these points across the image plays a crucial role in determining the quality of the result. If the points are overly concentrated, it becomes challenging to calculate the correct transformation, rendering the output unusable.

A transformation matrix is typically utilized to rotate an image by a specific angle. It is commonly represented in the form:

```
[cos(theta)  -sin(theta)]
[sin(theta)   cos(theta)]
```

This matrix facilitates the transformation of the second image, aligning it with the corresponding landmark points in the first image. By applying this technique to a time series of patient images, we can visualize and analyze the progress and changes in vitiligo treatment over time. Face overlay presents a valuable tool for monitoring the effectiveness of treatments and facilitating more informed decision-making in the field of vitiligo research.

To rotate an image by a certain angle θ by defining a transformation matrix M. This matrix is usually of the form:

$$M = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix}$$

OpenCV provides the ability to define the center of rotation for the image and a scale factor to resize the image as well. In that case, the transformation matrix gets modified.

$$\begin{bmatrix} \alpha & \beta & (1-\alpha)\cdot c_x - \beta \cdot c_y \\ -\beta & \alpha & \beta \cdot c_x + (1-\alpha)\cdot c_y \end{bmatrix}$$

In the above matrix:

$$\alpha = scale \cdot \cos\theta$$
$$\beta = scale \cdot \sin\theta$$

where c_x & c_y are the coordinates along which the image is rotated.

With the location of the mark points it is now possible to calculate a transformation and resize matrix.

The background detection is the first part if we can separate the background we can now take a more detailed look on the foreground. This will be discussed in the next section and will detail more object detection.

Hand Localisation and Extraction in Vitiligo Research

Hand detection in images is the process of identifying and localizing the presence of hands in an image. It is a popular task in computer vision and has various applications, such as gesture recognition, sign language recognition, and human-

computer interaction. Hand detection algorithms typically use techniques such as color segmentation, edge detection, and machine learning to locate and classify hand regions in an image. Some popular hand detection algorithms include Haar Cascades, HOG (Histogram of Oriented Gradients), and CNNs (Convolutional Neural Networks).

To detect hand and gestures there are some packages available in openCV also some KI models. The challenges in vitiligo are again the different patterns that are not trained in the models and therefore force wrong results.

MediaPipe is an open-source, cross-platform framework developed by Google that offers customizable machine-learning solutions for mobile, desktop, and web applications. It enables developers to build multimedia pipelines for processing and analyzing image, video, and audio data. MediaPipe provides a variety of pre-built, optimized models for tasks such as hand tracking, face detection, pose estimation, and object detection, which can be easily integrated into applications. Additionally, MediaPipe offers a set of tools for building and optimizing custom models using TensorFlow, TFLite, and other machine-learning frameworks.

The newer Mediapipe Project from Google has some more advanced trained models available. For the hand gesture the results also for Vitiligo are quite good. Trained with about 30k Images and using 3D waypoints as well as different backgrounds the models got quite robust. With this amount of data, the creation of self-trained models with the same quality is difficult or impossible. Those models work for general images but some occasions might occur where a specially trained model is required. As long as we can get sufficient results with the pre-trained models the effort to create new models is not recommended.

The Model can be set to have an accuracy of 70 % and to select a maximum of two hands per image for the patient images that is sufficient. The results show a good match of the hands in the image and therefore a location of the hands is possible. In Figure 6A) the network is not correctly matching the hands as the key point is selected on a different hand. This is again a limitation of the networks that are mostly trained with different datasets.

For the patient images the gesture classification is not relevant we are more focused on the hand location in the image. Getting the location points for the fingers etc we can build a bounding box around the area and crop to be able to continue vitiligo analysis solely for the hands.

With the following hand gestures classes:
['okay', 'peace', 'thumbs up', 'thumbs down', 'call me', 'stop', 'rock', 'live long', 'fist', ‘smile']

The model predicts:

1/1 [==============================] - 0s 78ms/step
[[9.46551140e-08 2.23430988e-13 5.38381115e-30 3.59471342e-05
  1.24289325e-24 **9.93946493e-01** 1.71923773e-07 6.01730589e-03
  5.89348969e-27 3.42078458e-17]]

1/1 [==============================] - 0s 20ms/step
[[2.4912276e-20 1.6994599e-12 0.0000000e+00 **9.9979144e-01** 1.4630577e-23
  1.5072704e-08 4.0178580e-10 2.0852947e-04 6.4788143e-33 4.7992202e-20]]

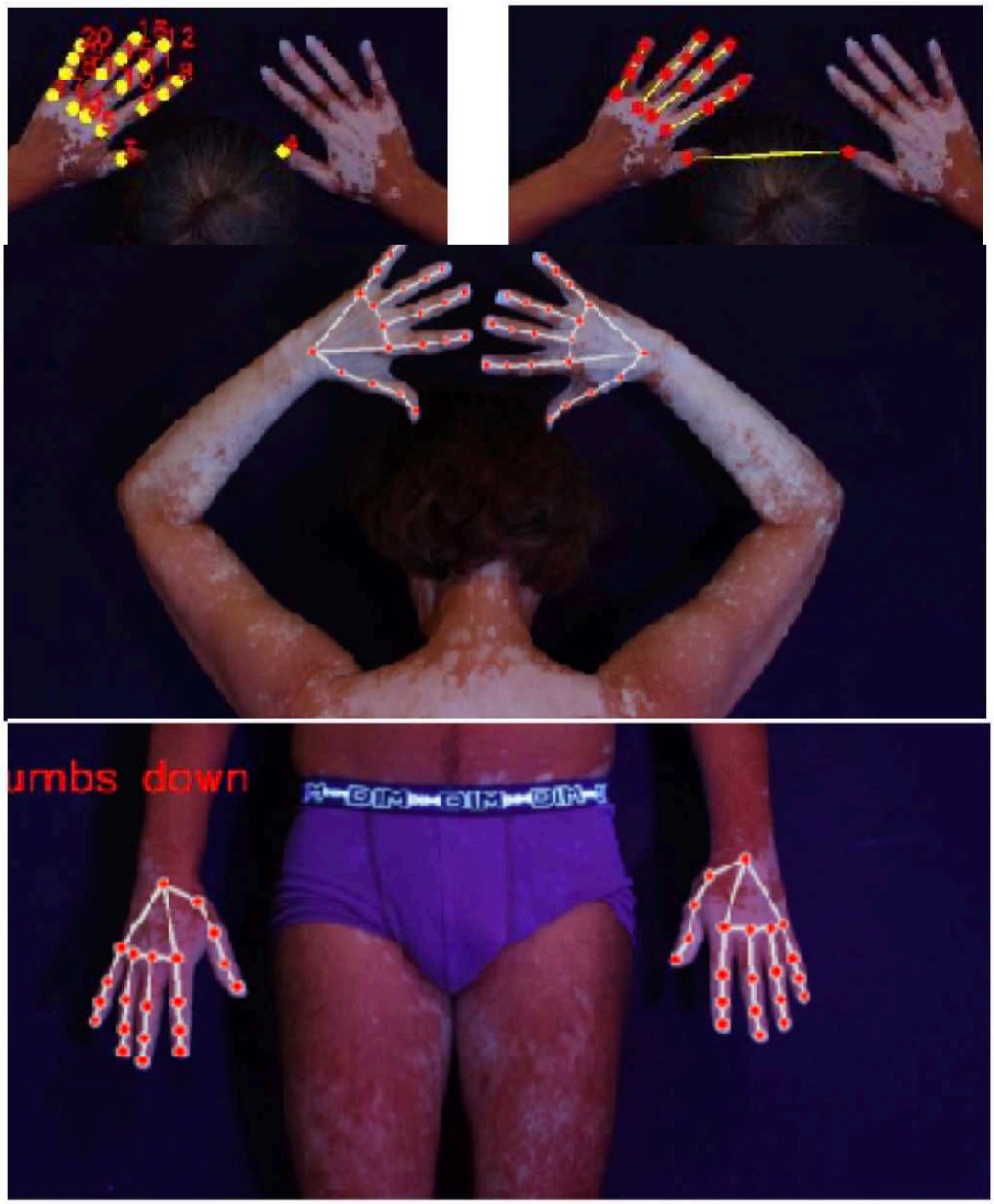
Thumbs down demonstrated in Figure 6B)

Figure 6) Hand Location on Vitiligo Images

A)

B)

C)



A) False Hand Detection B) Correct Hand Detection C) Gesture Detection

In the second example, the hands contain the vitiligo region and are matched also the gesture is correctly recognized. The information about the network performance is as follows:

['okay', 'peace', 'thumbs up', 'thumbs down', 'call me', 'stop', 'rock', 'live long', 'fist', ‘smile']

1/1 [==============================] - 0s 86ms/step
[[4.0570943e-17 3.0891023e-09 6.6938193e-09 7.1768200e-01 2.8230980e-01
7.6457496e-31 **9.0503972e-13** 2.2560296e-17 8.2333772e-06 1.1453810e-08]]

Step

1/1 [==============================] - 0s 18ms/step
[[1.9321494e-23 6.4738720e-07 7.3643676e-16 **7.4240524e-01** 2.5759417e-01
2.7913356e-35 1.4368196e-14 3.0874758e-13 5.4340896e-13 2.6790697e-10]]
Thumbs Down this is demonstrate in Figure 6C)

With this information, we can gather information about the location of the hand e.g. left or right. This classification is not necessary but might be helpful in case we like to create a template and fill the structure with images the position of the images should be known to place them in the right spot. This information can also be gathered by use of the model. The orientation of the thumb will indicate the location of the hand.

When the information is collected we get a series of points that will be connected. Each point returned by the model indicates a specific part of the hand.

To create a bounding box around those we need to find the maximum and minimum points which will define the area the bounding box needs to cover. The information about the hand if left or right is flipped since the viewpoint is different.

Detection of hands in the image is, therefore, possible and can be applied to the patient images to detect the ones that contain hands and to be used to separate and crop parts of the image for further processing. What about the rest of the body? We need to have a possibility to detect the foreground and separate the background.

Morphological Operations in Vitiligo Research

Erosion and dilation are two basic morphological operations in image processing. These operations are used to alter the shape and size of the objects in an image.

Erosion is a morphological operation that involves the removal of the outermost layer of pixels of an object in an image. This is done by sliding a small, predefined shape called a structuring element over the image and replacing the center pixel of the structuring element with the minimum pixel value in the area of the image covered by

the structuring element. This process results in the shrinkage of the object and the removal of small details and noise.

Dilation is the opposite of erosion, where the object in an image is expanded by adding pixels to its boundary. This is also done by sliding the structuring element over the image and replacing the center pixel of the structuring element with the maximum pixel value in the area of the image covered by the structuring element. This process results in the enlargement of the object and the filling of gaps in the object.

Erosion and dilation operations are often used together in a process called morphological opening and closing to smooth and enhance the objects in an image.

The process is currently only available for single-channel images means either a greyscale image or one channel for a BRG image.

The dilation works with a Kernel Matrix of uneven numbers the following algorithm is used:

1. A kernel(a matrix of odd size(3,5,7) is convolved with the image
2. A pixel element in the original image is ‘1’ if at least one pixel under the kernel is ‘1’.
3. It increases the white region in the image or the size of the foreground object increases

The erosion works in the other way the information is condensed only the most prominent information will be kept.

1. A kernel (a matrix of odd size e.g. 3,5,7..) is convolved with the image.
2. A pixel in the original image (either 1 or 0) will be considered 1 only if all the pixels under the kernel are 1, otherwise, it is eroded (made to zero).
3. Thus all the pixels near the boundary will be discarded depending upon the size of the kernel.
4. So the thickness or size of the foreground object decreases or simply the white region decreases in the image.

Most of the implementations today use erosion and deletion on the back and white images or mask images to locate the main objects in an image. The mask therefore is a threshold of some pixels or information located in an image. If the focus is on those objects we can with an erosion eliminate the minor objects detected. After the elimination of the minor object, the dilation of the remaining items will present a mask that contains the main objects in the image based on the applied filter.

This leads to the major topic the location of the correct threshold. Locating a well-suited threshold is the critical point discovered and where we had to divert into thresholding in more detail. How to find the correct threshold will be discussed in the coming sections.

Thresholding in Vitiligo Research

Thresholding is a technique in image processing and computer vision that involves converting a grayscale or color image into a binary image by setting a threshold value [57]. The threshold value separates the image pixels into two categories based on their intensity values: those that are below the threshold are set to zero (black), while those above the threshold are set to a maximum value (white). This process is commonly used for image segmentation, edge detection, and feature extraction in image analysis applications. Different thresholding methods exist, including global thresholding, adaptive thresholding, and Otsu's thresholding.

Finding the boundaries or limits is not only a mathematical problem. It is a major problem for all image processing. Where does an object of interest start and end? This is critical to object detection etc. When we could calculate the correct threshold we could easily extract the object of interest in an image. Multiple thresholding algorithms are available. They all work on single-channel images meaning grey or a single channel of a BGR image.

Binary Thresholding

This threshold function will set the result pixel to 255 (white) in case the pixel intensity > set threshold and to 0 (black) in case of a lower value

Binary Inverted Thresholding

This is the opposite case of Binary Thresholding

To zero Thresholding

The threshold function will set the resulting pixel to 0 (black) when the pixel intensity < set threshold value in the other case it will set it to 255 (white)

To zero Inverted Thresholding

The threshold function will set the opposite case of zero Thresholding

Truncated Thresholding

The threshold function will set the result pixel to the threshold if pixel intensity > threshold and in the other case the pixel values are set to be the same as the threshold and all other values remain the same.

Those five Thresholding algorithms are available and can be used within the programming libraries of Python. The unknown variable is always the threshold itself. If we find the correct threshold we would be finished and all problems would be solved. There is no best threshold algorithm the usage depends on the

circumstances and the input information. A more advanced algorithm working with stochastic will be discussed in the coming section.

Multi Otsu Threshold

Another approach is the multilevel thresholding. The multi-Otsu threshold [1] is a thresholding algorithm that is used to separate the pixels of an input image into several different classes, each one obtained according to the intensity of the grey levels within the image.

Multi-Otsu thresholding is a variation of Otsu's thresholding algorithm that can be used to segment images with multiple intensity peaks into multiple classes. Otsu's thresholding is a popular algorithm for automatic thresholding, which calculates a single threshold value to separate a grayscale image into two classes (foreground and background). In contrast, Multi-Otsu thresholding calculates multiple threshold values to segment an image into multiple classes, where each class corresponds to a different range of intensity values.

Multi-Otsu thresholding is particularly useful when the image has multiple objects with different intensities or when there is a need to segment an image into more than two classes. The algorithm works by iteratively finding the optimal threshold values that minimize the intra-class variance and maximize the inter-class variance for each class. The number of threshold values to be calculated depends on the number of intensity peaks in the image.

Multi-Otsu thresholding can be applied to a wide range of image analysis applications, including object recognition, medical image segmentation, and industrial inspection. It can help improve the accuracy and efficiency of image segmentation by automatically selecting optimal threshold values for each class.

Multi-Otsu calculates several thresholds, determined by the number of desired classes. The default number of classes is 3, for obtaining three classes, the algorithm returns two threshold values. First a lower bounding and second an upper bounding.
The buckets used within the algorithms are stochastically computed. The boundaries are set based on the values of the pixels.

Otsu reference proposes a criterion for maximizing the between-class variance of pixel intensity to perform picture thresholding. However, Otsu's formulation of the between-class variance is vital for the threshold calculation. Computation for the optimal thresholds of an image is difficult. First, the criterion of maximizing the usual between-class variance is proposed for image segmentation. Next, by the criterion, a recursive algorithm is designed to find the optimal threshold. Due to the Gaussian

distribution, Otsu works best on a binary distribution. If the image is black and white like writing on the image the threshold can be calculated very accurately. In case of a bias of the background, we will get incorrect thresholds. The way to overcome such problems would be to create a moving window or to separate the image into sections. With the use of histogram data, the pixels with the same value can be distributed widely in the image. This requires the data and the background to stay the same. In most images, the lightning is not ideal which leads to not equally distributed data and background information. For spot lighting the edges of the image would be wrongly considered as background since the background threshold for the center is already higher than the data value on the edges.

Otsu's thresholding method is a technique used for image segmentation. It calculates an optimal threshold to separate pixels into two classes, usually foreground, and background, based on intensity values.

Mathematical Formulation:

Given:
- Image histogram h(i) for intensity values i
- Total number of pixels N

The goal is to find a threshold T that maximizes the between-class variance σ_B^2 between foreground and background classes:

T = arg max_T { σ_B^2(T) }

Between-Class Variance:

σ_B^2(T) = w_0(T) * w_1(T) * (μ_0(T) - μ_1(T))^2

Where:
- w_0(T) and w_1(T) are probabilities of background and foreground classes
- μ_0(T) and μ_1(T) are means of background and foreground classes

Calculating Probabilities:
w_0(T) = ∑ (i=0 to T) h(i) / N
w_1(T) = ∑ (i=T+1 to L-1) h(i) / N
(L is the maximum intensity level)

Calculating Means:
μ_0(T) = (1 / w_0(T)) * ∑ (i=0 to T) i * h(i) / N

$\mu_1(T) = (1 / w_1(T)) * \sum (i=T+1 \text{ to } L-1)\ i * h(i) / N$

Implementation:
1. Calculate probabilities $w_0(T)$ and $w_1(T)$.
2. Calculate means $\mu_0(T)$ and $\mu_1(T)$.
3. Iterate through possible thresholds to find T that maximizes $\sigma_B^2(T)$.
4. Use the selected threshold T to binarize the image for segmentation.

We can imagine the histogram of an image to be separated into different buckets the limit until when a pixel belongs to which bucket is a stochastic problem and is only calculated by minimizing the variance within the buckets.

Again we are facing the limit problem also here. Finding a threshold allows us to locate objects and therefore their shapes will be of interest for further classification. For images that have different lightning, the thresholds vary from area to area in the image. Also, other features can be taken into account to find the correct thresholds. This leads to edge detection algorithms that allow the calculation of contours in an image by applying a transformation matrix to the image. Again the threshold values are problematic to discover. With relation to vitiligo, the threshold discovered should align with the vitiligo patches discovered. Where vitiligo is defined as to have in some regards sharp edges edge detection gets more into the scope and will be discussed in the coming sections in regard or threshold location and additional edge information.

Edge Detection in Vitiligo Research

Some vitiligo defines skin without pigmentation and therefore has a sharp border with the pigmented skin. One indication of vitiligo is to present sharp edges with the normal skin. Therefore we could try to use algorithms that allow us to find or locate contours or edges in an image. For this kind of analysis, mostly one channel or grey images are used. We can use the blue channel since this is the most prominent information in an image. When we use edge detection the information is calculated and we can show the edges with an overlay in the original image. If we look at the skin the image contains a different lighting in all areas of the image. The light is coming from one direction the skin values differ, therefore. As shown above, we get a lot of information about edges that are not relevant to vitiligo but might be differences in normal skin.

Edge Detection is a technique used to detect objects it depends as most techniques on the right threshold. OpenCV provides an implementation that can be used in Python programs after importing the CV2 Library.

cv2.canny() is a function provided by OpenCV, a computer vision library. It is used to detect edges in an input image using the Canny edge detection algorithm.
The Canny edge detection algorithm is a multi-stage process that works as follows:

1. **Noise Reduction:** Since edge detection is susceptible to noise in the image, the first step is to remove any noise from the image using a Gaussian filter.

2. **Gradient Calculation:** The next step is to calculate the gradient of the smoothed image using the Sobel operator. This is done to highlight the regions of the image with high spatial gradients, which often correspond to edges.

3. **Non-maximum Suppression:** This step involves thinning the edges obtained from the gradient calculation to a single pixel width. This is done by suppressing all non-maximum pixel values, which helps to sharpen the edges.

4. **Double Thresholding:** In this step, two thresholds are defined - a low threshold and a high threshold. Any edge pixel value that is above the high threshold is considered a strong edge, while any pixel value below the low threshold is considered a non-edge. Pixels with values between the low and high thresholds are considered weak edges.

5. **Edge Tracking by Hysteresis:** The final step involves tracking the edges by hysteresis, which is a technique for filtering out weak edges that are not connected to strong edges. Any weak edge that is adjacent to a strong edge is considered part of the edge, while weak edges that are not adjacent to strong edges are discarded.

The cv2.canny() function takes the input image as its first argument, and the low and high thresholds as its second and third arguments, respectively. It returns a binary edge map, where the edges are marked with white pixels and the non-edges are marked with black pixels. The non-pigmented skin should create an edge when we have a sharp color break.

Canny requires thresholds or upper and lower limits. This leads to the problem of finding the correct settings.
Determining the best canny upper and lower limits for an HSV image depends on the specific image and the edge detection requirements of the application. Typically, the appropriate thresholds are chosen through experimentation and trial and error.
When using Canny edge detection with an HSV image, the choice of which channel to use (Hue, Saturation, or Value) depends on the specific characteristics of the

image and the edges of interest. Here are some general guidelines:

- **Hue (H) channel:** The hue channel is useful for detecting edges based on color changes. For example, if the edges of interest correspond to a change in color in the image, the hue channel can be used for edge detection.

- **Saturation (S) channel:** The saturation channel is useful for detecting edges based on changes in color intensity. For example, if the edges of interest correspond to changes in color saturation, the saturation channel can be used for edge detection.

- **Value (V) channel:** The value channel is useful for detecting edges based on changes in brightness or intensity. For example, if the edges of interest correspond to changes in brightness or shadow in the image, the value channel can be used for edge detection.

As for the choice of upper and lower limits, it is important to balance the sensitivity and specificity of the edge detection. A lower threshold value will detect more edges, but may also include noise and false positives. A higher threshold value will be more selective, but may also miss some edges.

Here are some general guidelines to consider when selecting the upper and lower limits:

- **Lower threshold:** A lower threshold of 10-20 is a good starting point to detect most of the edges in the image. However, the optimal lower threshold may vary depending on the specific image and the edge detection requirements.

- **Upper threshold:** The upper threshold is usually set to a multiple (2-3 times) of the lower threshold. This helps to filter out weaker edges and reduce false positives.

Again, these are just general guidelines, and the best values for the thresholds depend on the specific image and the edge detection requirements. It is required to experiment with different values and visually inspect the results to find the optimal values.

Edge detection in Computer vision is used to detect objects we will take a look at available edge detection algorithms and will discuss the best use case for each. When writing or buildings with sharp edges are the objects of interest edge detection is a helpful tool to use.

Edge detection is a fundamental concept in image processing that involves identifying the boundaries between objects and backgrounds in an image. The edges are defined as the points where the brightness of an image changes abruptly or has a discontinuity. Edge detection is useful in various applications, including image segmentation, object detection, and feature extraction. There are several methods for edge detection, including the Sobel, Canny, and Roberts cross methods, among others. These methods typically involve applying a filter to the image and identifying the regions with the highest gradient or change in intensity, which corresponds to the edges in the image.

Sobel Feldman

Sobel-Feldman edge detection is a widely used method for detecting edges in an image. It is a gradient-based method that calculates the first derivative of the image intensity at each pixel position. The Sobel-Feldman operator uses two 3x3 convolution filters to estimate the horizontal and vertical gradient components of the image at each pixel position.

The horizontal and vertical gradient components are then combined to obtain an estimate of the edge strength and orientation at each pixel. This method is particularly useful for detecting edges in noisy images, as the filtering process helps to reduce noise before the edge detection is performed. The Sobel-Feldman operator is named after its inventors, Irwin Sobel and Gary Feldman, who published the method in 1968.

Edge detection algorithms are available as part of the OpenCV Python Library. Based on C and C++ libraries are. The edge detectors that are also commonly used are Sobel Feldman and Roberts Cross. Whereas Sobel uses a matrix that extracts horizontal as well as vertical pixels that are close together. Sobel masks use a design that provides maximum value at horizontal or vertical *edge orientation*. Both masks have the same coefficients as they are rotated 90 degrees to each other.

Gx = [[+1, 0,-1][+2, 0,-2][+1, 0,-1]] *I;

Gy = [[+1, +2,+1][0, 0,0][-1, -2,-1]] *I;

Improving Images with Parable Function

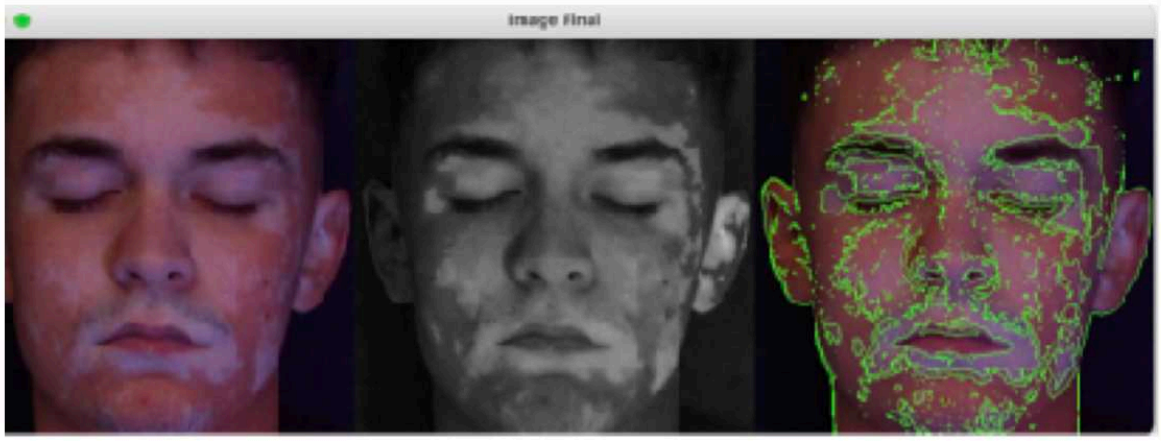
When we look at the face the side starting at the eyes is less exposed to the light the area is in general darker. The idea to improve the sides to get better results lead to the use of a parable. Applying a stretched parable over the image allowed it to

increase the side values for the pixels with higher values than the middle of the image. In Figure 7A) we can see the original image and the contours that were generated. We can see on the left side the shade is removing the contours in that area. In Figure 7B) we can see the image adjusted with the parable values which improves the lightning situation so that also on the left side of the face the shades disappear and the contours are more present.

Figure 7) Edge Detection in Vitiligo Images

A)

B)



A) Image with Edge Detection B) Image with Parable Adoption and Edge Detection

The face of a human is only a short part of the face. The side continues quite steep making it for the light difficult to reach the skin at the same time as on the front which results in less light disposal in that area.

If we place a parable on the face with the 0 point as the nose line we can adjust the pixel values according to the value of the parable and make the image lighter on the

outside. The parable formula is to be audited to the image size but since we can resize the image to a certain number we can use a constant to construct the parable.

Roberts Cross in Vitiligo Research

The other variants Robert Cross have less accurate results for horizontal and vertical lines but get more information for other line orientations. The calculation is in the same way as the Sobel calculation but with a different kernel.

Gx = [[+1 0][0 -1]] ; Gy = [[0 +1][-1 0]]

The information the Robert Cross kernel is returning is not very informative for the discovery of vitiligo edges. The information is for a human eye not very informative a mapping or overlay and locating possible edges is not possible.

Another field is skin detection. Human skin has a vast distribution in the color scheme. Depending on the ethnical origin the human skin can be pale white for an Inuit or relatively dark and brown for an African. With this range of skin tones and its variations skin detection is a difficult area. The information we can obtain from the surface of the human skin is not solid since all pixels differ in a certain way due to the light or the skin color itself. This leads to a fragmented result image that contains non-vital information that can not be used for further processing. The pixels are a combination of the x and y matrix results and are combined into the below result image.

Laplacian Edge Detection in Vitiligo Research

Laplacian edge detection is an image processing technique used to detect edges or boundaries in an image. It is based on the second derivative of the image intensity. The Laplacian operator is a mathematical operator that computes the Laplacian of an image, which represents the rate of change of the image intensity.

The Laplacian operator is typically applied to an image by convolving the image with a Laplacian kernel, such as:

```
[[0, 1, 0],
 [1, -4, 1],
 [0, 1, 0]]
```

The Laplacian kernel is a discrete approximation of the second derivative, and it highlights regions in the image where the intensity changes rapidly, indicating the presence of edges.

When the Laplacian operator is applied to an image, the positive values in the output indicate light-to-dark intensity transitions, while the negative values indicate dark-to-light transitions. The zero-crossings in the output represent the locations of edges in the image.

To obtain a binary edge map, a threshold is applied to the output of the Laplacian operator. Values above the threshold are considered as edges, and values below the threshold are considered as non-edges.

Fast Fourier Transformation in Vitiligo Research

We can use Fast Fourier Transformation (FFT) on images. The FFT is mostly used in signal processing the information provided out of an FFT is information about the time.

If we relate that to a single image, we have the image taken at a certain time. The objects in this image might have been moved or were in movement when the image was taken. We will then see the center for the FFT to be on the moving object. If we look from this center and relate all other points to this as a fixed point we will then see the background moving about the discovered centre point.

We can therefore locate a moving object in an image or detect the blur in an image. Since the background or the moving object is not in focus for the time it took to take the image.

There are use cases where we like to find sharp contours like handwriting on paper or houses in an image. With very sharp contours we can get quite good results when applying one of the above methods. If we take a look at a bit more fuzzy information like soft edges or not that much contrast available or when we can’t find a correct threshold in an image the results tend to be less valuable.

Template Matching

One potential approach for object detection is template matching, which involves comparing a template image, representing the object of interest, with different regions of the input image to identify matching patterns. In template matching, the template serves as a fixed reference to locate the position of the object within the image.

The process begins by patching the input image with the template and comparing the similarity between the template and each patched region. When a match is found, a bounding box or a similarity score can be returned, indicating the location of the object. Template matching is particularly useful when the object to be detected

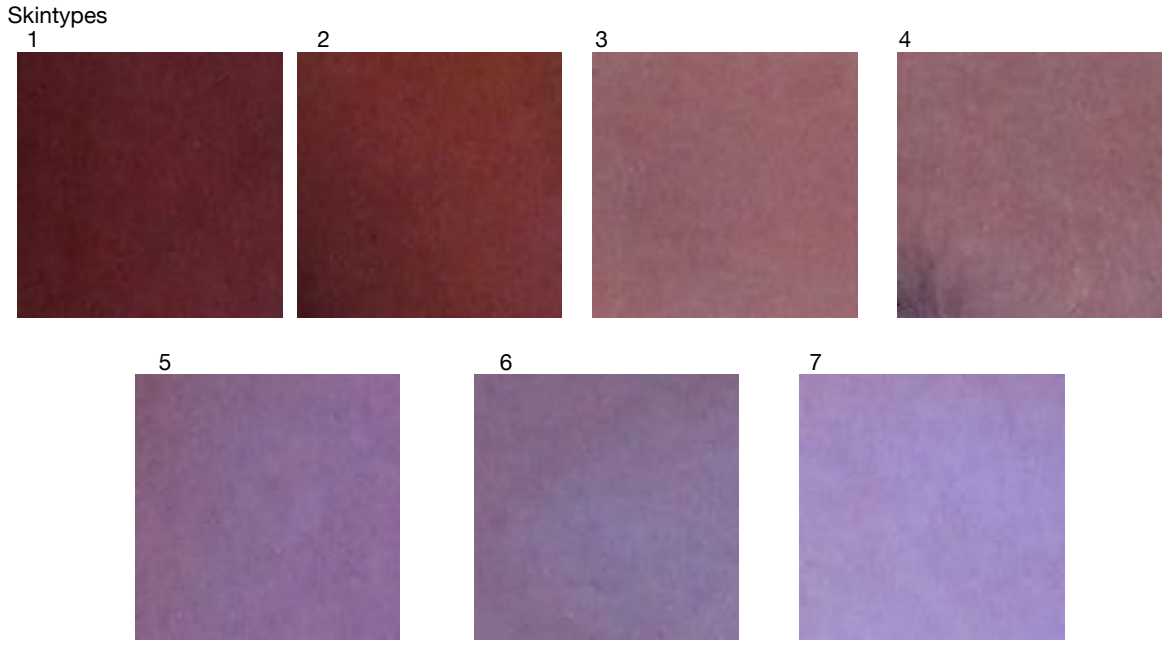
remains relatively consistent in size, shape, and appearance across different instances.

Histogram Back Backtracking in Vitiligo Research

The idea behind histogram backtracking is to use the histogram of a skin patch and check for its occurrence in the original image. This provides a means to identify the predominant skin tone in the image or if the vitiligo regions can be located.
When using skin patches of different skin temperatures we can locate the areas in the image and therefore create a mask.

Skin patches ranging from dark to light colors were created Figure 8), and the histogram backtracking technique was applied to detect the presence of specific skin tones in the image. However, determining the correct skin type can be challenging. For example, if an image contains a significant amount of non-pigmented skin, the

Figure 8) Skin Types



1) (hMin = 167 , sMin = 139, vMin = 69), (hMax = 185 , sMax = 203, vMax = 104)
2) (hMin = 0 , sMin = 130, vMin = 62), (hMax = 182 , sMax = 183, vMax = 132)
3) (hMin = 170 , sMin = 69, vMin = 128), (hMax = 184 , sMax = 100, vMax = 171)
4) (hMin = 161 , sMin = 61, vMin = 62), (hMax = 184 , sMax = 92, vMax = 165)
5) (hMin = 125 , sMin = 46, vMin = 126), (hMax = 156 , sMax = 74, vMax = 170)
6) (hMin = 135 , sMin = 54, vMin = 62) , (hMax = 162 , sMax = 88, vMax = 167)
7) (hMin = 121 , sMin = 56, vMin = 171), (hMax = 140 , sMax = 86, vMax = 214)

presence of darker skin tones may be limited. In such cases, simply calculating the number of returned pixels is not sufficient.

The following algorithm was implemented:

1. Define the skin patches ranging from dark to light colors.

2. Convert the image to the HSV color space.

3. Calculate the histogram of the HSV image using the selected skin patch.

4. Find the maximum value in the histogram.

5. Determine the skin type based on the peak value in the histogram.

6. Use the identified skin type to select the appropriate trained network for segmentation.

When applying the patches in an order from dark to light we can locate the pigmented skin up to the non-pigmented skin in the skin types defined.

Vitiligo Amount Calculations in Vitiligo Research

The question we like to answer is did the vitiligo treatment help did it improve the patient's health? Can we have an objective number that indicates the current state? Determining the exact amount of vitiligo in an image presents a challenge due to the absence of a reference measurement or scale. In clinical settings, the estimation of vitiligo extent is often done by comparing it to the size of a thumb or using fingers to indicate a percentage. However, these methods lack precision and do not provide accurate measurements in square centimeters of the non-pigmented skin area.
In the absence of a reliable measurement system within the image itself, alternative approaches need to be considered. One potential approach involved utilizing the assumption that the size of the eyelid remains relatively constant across individuals. The variability in face poses, expressions, and image quality further complicated the task of accurately determining the vitiligo measurement. Despite these challenges, efforts were made to develop techniques and algorithms that could provide reasonably accurate estimations of vitiligo extent in the absence of a reference measurement.

Considering the limitations and challenges in accurately measuring the extent of vitiligo in an image, a practical approach was adopted. The total area of the complete skin visible in the image was taken as 100%, and the number of non-pigmented

pixels detected was measured as a percentage of this total. This methodology provided a reasonably precise estimation of the non-pigmented area within the image.

While it may not provide an exact measurement in terms of square centimeters, this approach offers a practical and reliable means of quantifying the extent of vitiligo present in the image. By leveraging the pixel-based analysis, the algorithm provides valuable insights into the distribution and proportion of non-pigmented regions, contributing to a comprehensive understanding of the vitiligo condition.

Machine Learning in Vitiligo Research

Machine Learning offers the opportunity to get algorithms implemented that are way too complex or have too many conditions to check so that a program can not cope with the requirements or gets way too complex. Here we can utilize the area of Machine Learning. A network of neutrons is connected and weights are trained and if the training is accurate enough we get an input and output with a probability.

Machine Learning contains 4 different areas:

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Semi-Supervised Learning

Supervised Learning

With supervised learning images will be tagged with different labels and can then be used in training since the information and correctness can be proven. The overhead is the generation of the data and the effort it takes to create the different data classes.

Unsupervised Learning

Unsupervised Learning is the opposite approach. A set of images are provided and the goal is to fit them into different categories depending on features that are in common.

Reinforcement Learning

The idea is to discover what the best result is. Based on decisions taken a cost function will indicate the quality if the decision is wrong the function returns a higher number if the decision is correct a smaller number. This will lead to a learning curve that can be used to optimize the networks and improve the accuracy of the results.

Semi-supervised Learning

Semi-supervised kind of learning, the calculation is prepared upon a mix of named and unlabelled information. Normally, this blend will contain a limited quantity of named information and a lot of unlabelled information. The fundamental method included is that first a group of comparable information will be utilized for an unaided

learning calculation and afterward utilize the current named information to name the remainder of the unlabelled information.

The effectiveness of the U-net network in processing AI outcomes is intricately linked to the quality and quantity of training data. As a supervised learning model, the U-net network relies on annotated data to accurately segment images. This annotated data consists of pairs of input images and corresponding labeled masks, where each pixel in the mask is assigned a label indicating its class.

Insufficient or low-quality training data can impede the U-net network's ability to learn relevant features and patterns necessary for accurate segmentation. Overfitting, where the model performs well on training data but fails to generalize to unseen data, can result from such shortcomings.

To address these challenges and ensure robust segmentation results, it is imperative to curate a large and diverse training dataset. This dataset should encompass various conditions, including different image modalities, imaging protocols, and anatomical variations. Expert annotation of the training data is crucial to ensure accurate labels and minimize segmentation errors.

Furthermore, maintaining a balanced dataset with an adequate representation of each class is essential to avoid bias towards the dominant class. In medical imaging applications, imbalanced datasets can lead to inaccurate diagnostic or treatment decisions with potentially serious consequences.

Variations in lighting conditions necessitated the partitioning of images and separate processing of each section. This requirement implies the creation of multiple networks, each dedicated to a specific section. Additionally, differences in skin color posed a challenge, requiring a secondary separation, effectively doubling the number of required networks.

In summary, the quality, quantity, and diversity of training data are indispensable for the U-net network to learn the necessary features and patterns for accurate image segmentation. Overcoming challenges related to imbalanced datasets and variations in lighting conditions is essential for achieving optimal performance in vitiligo segmentation.

The basis for Machine Learning is data. This leads to the area of generating masks that can be used for training.

Generating Masks for Vitiligo Research

The generation of masks is crucial for the AI algorithm to generate training data. To generate this mask conventional computer graphics are tried. The results are presented in the following sections always with a focus on locating vitiligo.

Adaptive Mask Creation for the Head in Vitiligo Research

Stepping back and taking a new angle to tangle the problem again. We were focusing on the blue channel. What if we consider the complete image information? We will not be able to use some of the thresholding algorithms since they work only on one channel.

To generate masks for training the UNet model, the HSV (Hue/Saturation/Value) color scheme proves to be the most suitable. By adjusting the hue and saturation values, we can create masks that highlight the vitiligo regions within the image with a focus on the head.

To locate the correct saturation minimum, we observe that vitiligo areas exhibit a distinct decrease in saturation when the lighter parts of the image are encountered. By adjusting the minimum saturation value, we can precisely identify the boundary of the vitiligo regions.

However, the maximum saturation value varies depending on image conditions and skin tones. As discussed in the section on Histogram Backtracking, the following algorithm can be employed:

1. Convert the image to the HSV color space.

2. Adjust the saturation value to find the minimum value that captures the non-pigmented areas.

3. Determine the maximum saturation value based on the image conditions and skin tones.

4. Create a mask that highlights the vitiligo regions based on the adjusted saturation values.

By following this algorithm, we can effectively generate masks that accurately delineate the vitiligo regions, enabling robust training of the UNet model for vitiligo segmentation.

When we use a graphical user interface for several images to create manually a mask one can notice the following there is a standard procedure evolving. The image

needs to be split into three channels H, S, and V. If we take a look at the HSV image channels we can see some channels are more important in defining the skin values. We therefore create seven sliders, upper and lower values for H, S, and V and one for the option to save the results Figure 9). Those seven sliders allow us to easily adjust and generate masks.

Figure 9) HSV Slider with Original Image

We can find the lower S value quite easily, but only in the case that now white objects like hair or underwear are in the picture.

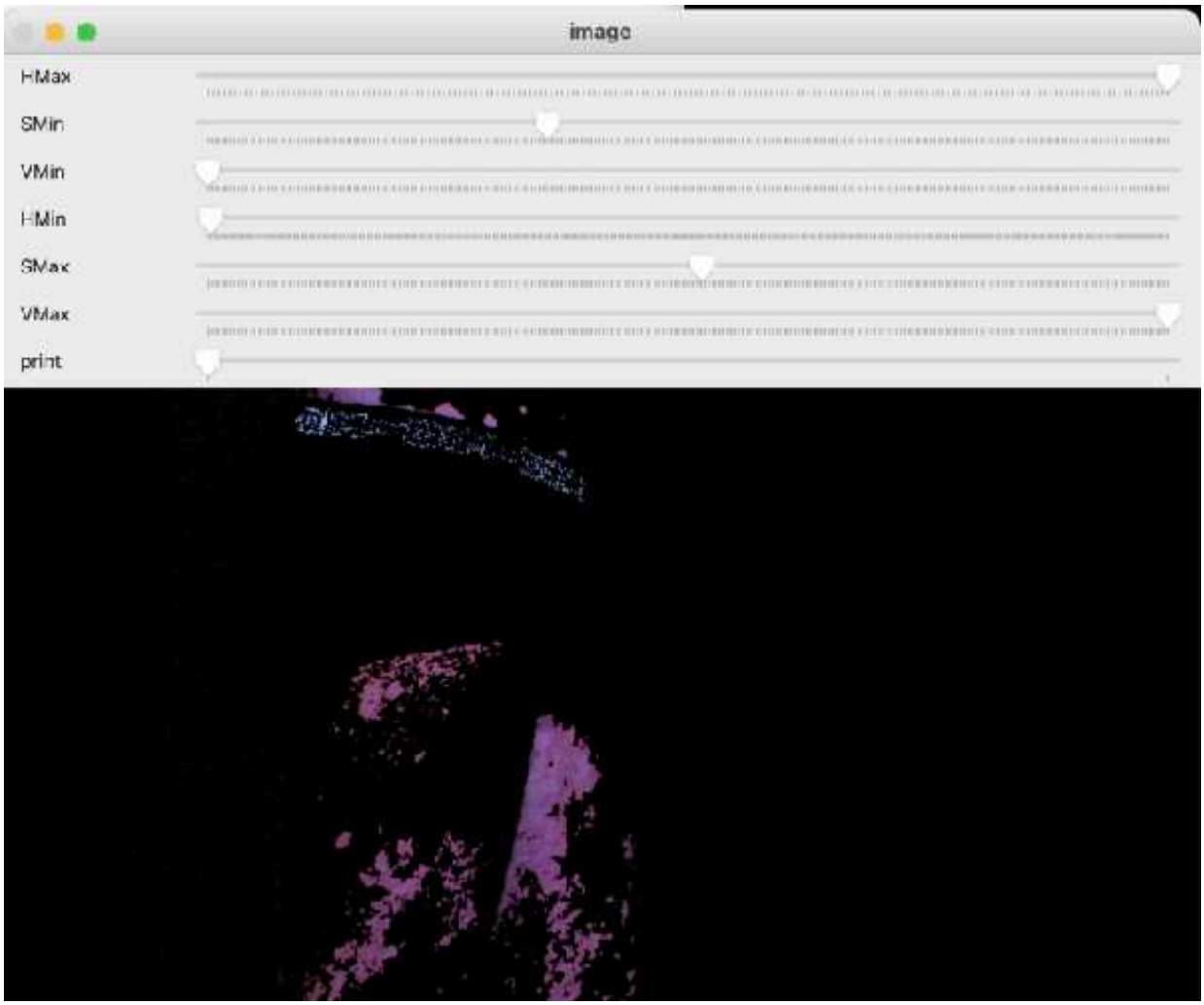
The lower S value is found when we lose the first pixels in the image containing only skin tones when slowly increasing the Smin value. We can use this value to locate the starting region for vitiligo in the image which would be the highest lowest S value. Since the saturation of the bright areas is the least. What also is obvious is that the lower the S value is the larger the distance between S lower and S upper value. Also

if S is lower in a range of around 30 the H upper value needs to be adjusted. This applies some complexity since both values are not calculable.

The upper H value serves as an indicator that influences the display of skin tones. If this value is set too low, no results are obtained. Determining an appropriate upper H value is crucial to avoid losing the pixels in the selected lower S values.

Similarly, the upper V value has a similar impact as the upper H value. If it is reduced excessively, desired results may not be achieved. Considering the lower S value helps establish a lower boundary for the upper V value, particularly when it comes to vitiligo-affected skin. On the other hand, increasing the lower V value gradually diminishes the skin area Figure 10).

Figure 10) Image with Changed HSV Slider for Vitiligo Detection



Therefore, it can be set to a specific level at which pixels in the image begin to disappear, providing a measurable threshold.

Determining an appropriate upper S value presents a unique challenge in the context of adaptive HSV adjustments for vitiligo detection. No universally applicable threshold definitively indicates when S has been reduced excessively. To address this challenge, we adopted a method that involves establishing an upper S value based on the S lower value. Specifically, we set the upper S value as S lower + 42 pixels. This approach allows us to create a dynamic range within which vitiligo-related color variations can be accommodated.

For a more granular understanding of this dynamic range, we turned to the HSV (Hue, Saturation, Value) color scheme and analyzed sample skin patches extracted from the provided images. Within these images, we identified seven distinct major skin colors, ranging from dark-pigmented skin to light non-pigmented skin. By closely examining the H and S values within these patches, we gleaned valuable insights into the suitable boundaries for adaptive HSV adjustments. These observations yielded important information about the starting and ending values for the H and S components of the color spectrum within each skin type.

This empirical approach enables us to define Smax values that vary depending on the specific skin tone present in the image. Consequently, we can tailor our adaptive HSV adjustments to account for the diversity of skin colors, improving the accuracy and robustness of our vitiligo detection methodology.

Ant Network

There are several Python libraries available that implement ant colony optimization, including the ACO library and the PyACO library.

To implement an ant colony optimization algorithm, you will need to define the problem you are trying to solve, the pheromone update rules, and the ant behavior rules. Here's a high-level overview of the steps involved in implementing an ant colony optimization algorithm:

1. Define the problem you want to solve and formulate it as an optimization problem.
2. Create a graph representation of the problem, where each node represents a possible solution and each edge represents the transition between solutions.
3. Initialize the pheromone levels on each edge to a small value.
4. Create a set of ants and define their behavior rules. Each should choose a path through the graph based on the pheromone levels and some heuristics that guide their exploration.

5. Evaluate the quality of each ant's solution and update the pheromone levels on the edges they traversed based on the quality of their solution.

6. Repeat steps 4 and 5 for a certain number of iterations or until a stopping criterion is met.

7. Select the best solution found by the ants as the final solution.

It's worth noting that while ant colony optimization can be a powerful optimization technique, it may not be the best choice for all problems. It's important to consider the specific characteristics of your problem and the performance of the algorithm on similar problems before deciding to use ant colony optimization.

Processing the image as an ant colony each cell is connected with the surrounding pixels. The idea to use a brain matrix to store information about a pixel whether it is vitiligo or not comes from the ants which work in the same way. The information is exchanged to the next ant which can then either be changed based on the new information. In programming, this can be accomplished by providing the information to the next neighbor a brain can be installed. Processing over this brain with a weight function leads to clusters of white areas forming vitiligo sections. The algorithm used can be described as follows:

1) initialize the ant network
2) Initialize the brain network
3) For each check neighbor ants and store the result in the brain
4) Repeat until all brain cells are set

Hill Algorithm Approach in Vitiligo Research

The hill climbing algorithm is a heuristic search algorithm that can be used for optimization problems. It is typically used for finding a local maximum or minimum of a function, by iteratively adjusting the parameters of the function in a way that leads to the best possible outcome. While hill climbing can be used for some image analysis tasks, it may not be the best approach for vitiligo detection, which typically involves more complex image analysis techniques.

One possible approach to using hill climbing for vitiligo detection would be to define a fitness function that measures the similarity between a given image patch and a reference patch of normal skin. The hill climbing algorithm could then be used to iteratively adjust the parameters of the image patch to maximize the similarity score with the reference patch.

Here's a rough outline of how you might approach vitiligo detection using the hill-climbing algorithm:

1. Collect a dataset of images of skin patches, some of which have vitiligo and some of which do not. You can either collect your own dataset or use an existing one such as the ISIC Archive.

2. Preprocess the images by resizing them to a uniform size and normalizing their pixel values.

3. Define a reference patch of normal skin, which could be a patch from a non-affected area of the same image or a different image.

4. Define a fitness function that measures the similarity between a given image patch and the reference patch. This could be based on pixel intensity values, texture features, or other descriptors.

5. Use the hill climbing algorithm to iteratively adjust the parameters of the image patch to maximize the similarity score with the reference patch. The algorithm should start with an initial image patch and make small random perturbations to the patch in each iteration.

6. Evaluate the final image patch to determine whether it contains vitiligo. This could be done by thresholding the similarity score or using a machine learning classifier to make the final decision.

While this approach may work for some cases, it is worth noting that vitiligo detection is a complex problem and may require more sophisticated image analysis techniques such as segmentation, feature extraction, and classification.

The idea is to use a more complete view of the image and then look at the skin temperatures as on a map indicating vitiligo areas with high values for blue as peaks. This when cutting the information from the bottom would lead to a topology card of the skin. The image was thresholded by a base value and then an XOR was used with the next layer. If the layer was an intersection we know the hill continued. The remaining pixels were therefore boundaries.

FloodFill in Vitiligo Research

Flood filling is a technique commonly used in image processing to identify and isolate connected regions of pixels in an image that share a common color or intensity value. While it can be used for some image analysis tasks, it may not be the best approach for vitiligo recognition in images, which typically requires more complex techniques.

One possible approach to using flood filling for vitiligo recognition would be to identify the regions of the image that have a different color or texture than the surrounding skin. This could involve segmenting the image into regions based on color or texture

features and then applying flood filling to each segmented region to isolate the regions that are affected by vitiligo.

A rough outline of an approach to vitiligo recognition using flood filling could be:

1. Collect a dataset of images of skin patches, some of which have vitiligo and some of which do not. You can either collect your own dataset or use an existing one such as the ISIC Archive.
2. Preprocess the images by resizing them to a uniform size and normalizing their pixel values.
3. Apply image segmentation techniques to separate the skin patches from the background. This could involve thresholding, edge detection, or other segmentation techniques.
4. Extract features from the segmented regions of the image, such as color histograms or texture features.
5. Use flood filling to isolate the regions of the image that have a different color or texture than the surrounding skin. This could involve selecting a seed pixel in each segmented region and recursively filling neighboring pixels that meet certain color or intensity criteria.
6. Evaluate the isolated regions to determine whether they contain vitiligo. This could be done by thresholding the size or intensity of the isolated regions, or by using a machine learning classifier to make the final decision.

Moving Thresholds in Vitiligo Research

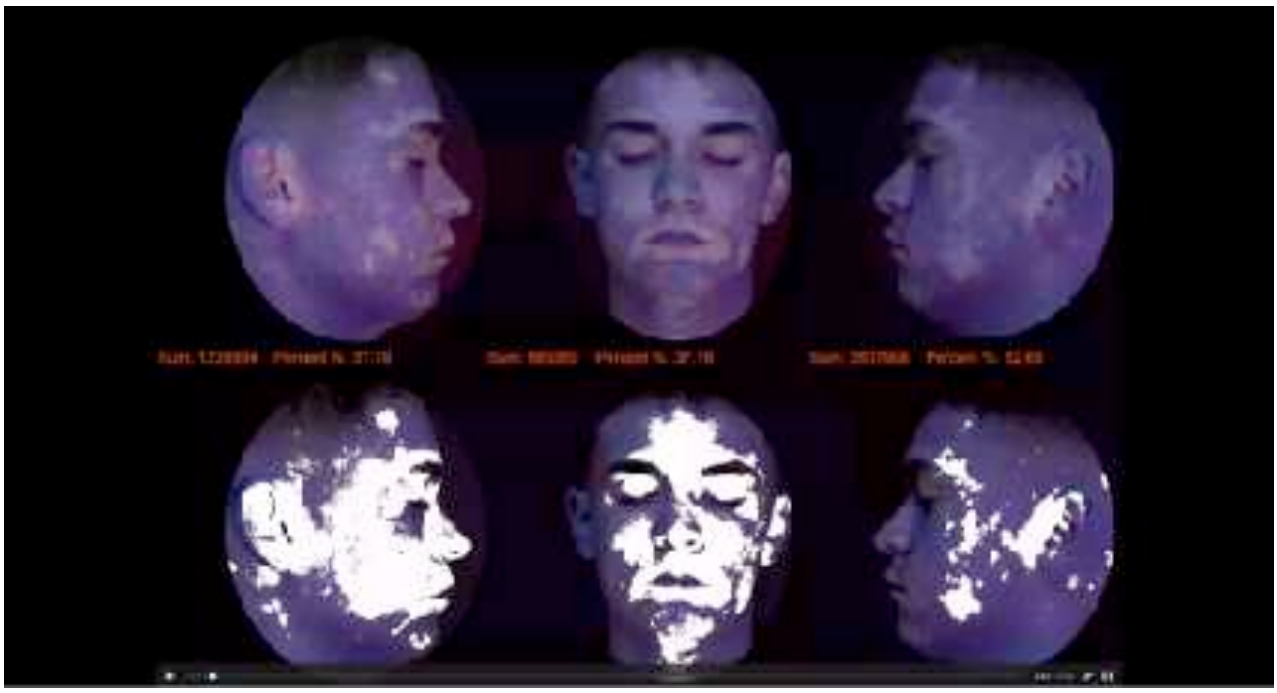
Moving thresholds can be used in vitiligo recognition to adaptively adjust the threshold values used for segmentation and feature extraction based on the local characteristics of the image. This can help to overcome some of the limitations of fixed thresholding approaches, which can be sensitive to variations in lighting, contrast, and other imaging conditions.

The idea is to create a video sequence with a moving threshold which would allow the physician to stop at a certain point and collect the correct information for the frame displayed.

The images would be processed and a mask created in which the count would be displayed. Each frame is combined into a movie file and saved in a movie format. The movie player provides the user interface for image processing.

The time is used to define the threshold the numbers are calculated. The images are processed in a complete block and lightning differences are not taken into account. The images contain the patient's face from both sides and the frontal view in the middle as shown in Figure 11). The thresholds are set for all images the same. This leads to different accuracy depending on the lightning and the correct setting for one image might not correspond to the correct values for the other image.

Figure 11) Moving Threshold Video



With the idea to have a time series movie created we could change the threshold for each frame that we create and concatenate those. With the slider proved in the movie player and a calculated number we can select a time and get the amount of vitiligo presented in the image.

Watershed Algorithm in Vitiligo Research

The watershed algorithm is a technique used in image processing to segment images into distinct regions based on their topology. It works by treating the image as a topographic map, where the brightness of each pixel represents its elevation, and then dividing the image into regions based on the locations of the "basins" or valleys in this map.

Image Overlay in Vitiligo Research

The idea to overlay two face images and to show the differences in the images led to problems in identifying correct waypoints. The transformations and shift algorithms locate features in the image to find the corresponding features in a different image and match the 2 Points.

When overlaying face images, it is important to ensure that the images are aligned properly to avoid artifacts such as double contours or misaligned facial features. This can be achieved by using image registration techniques to align the images based on corresponding facial landmarks or features.

Another important consideration is the lighting and color consistency between the images. If the lighting conditions and color balance are significantly different between the images, the overlaid result may look unnatural or unrealistic. In such cases, it may be necessary to apply color correction or tone mapping techniques to achieve a more seamless blending of the images.

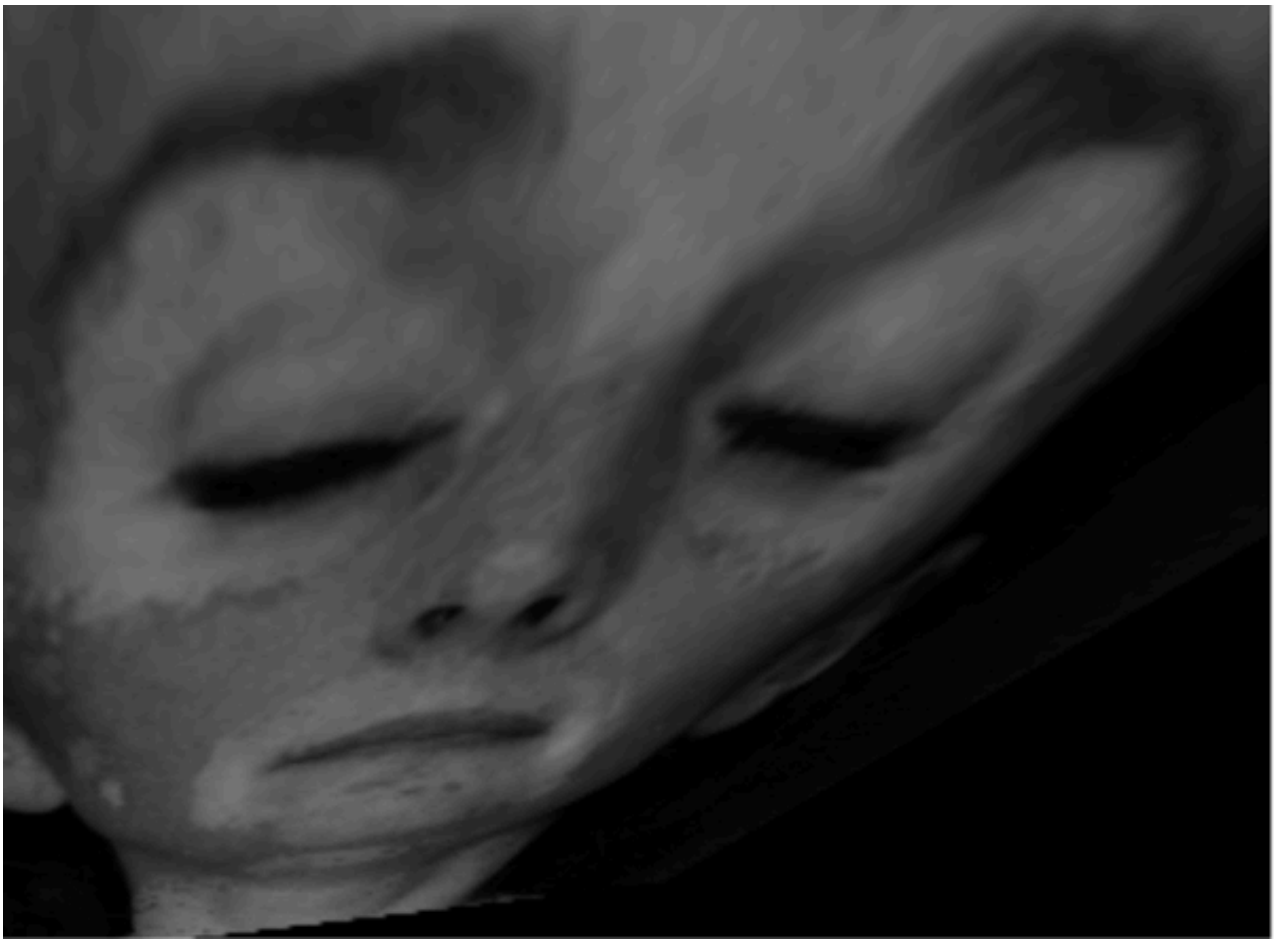
Overall, while it is technically possible to overlay two face images taken at different times and from different angles, the success of the operation will depend on the quality and compatibility of the images, as well as the accuracy of the image registration and blending techniques used. As shown in Figure 12) the points are not matched and the result is an image that was transformed in the wrong way.

Figure 12) False Image Overlay

When we want to overlay images we need to have at least five fix points. At least 5 points need to be found to create a transformation and transpose matrix. To locate those fixed points it is possible to run a landmark detection on the image. The landmark detections work great for objects that are kind of uniquely identified. The human head is the eyes nose mouth ears etc. When taking into account that the vitiligo patterns are also landmarks they complicate the situations even more. If the points in both images can not be aligned correctly or are different due to changed parameters e.g. different hair cuts the transformation can not be done correctly.

Blue Channel Processing in Vitiligo Research

Most of the image processing is due to the complexity based on grayscale images. The grayscale reduces the complexity to one level instead of three levels. The color information is not taken into account. That is also the case for threshold algorithms like CLAHE (Contrast Limited Adaptive Histogram Equalization) or thresholds. When looking at the grayscale image we are focusing on the blue channel because blue is the most prominent color channel in the images it has the most information available. This is due to the extra flashlight that is used when taking the images. Under normal



conditions, the blue channel in the image would still be prominent but this is caused since all colors have an amount of blue in them.

Contrast Limited Adaptive Histogram Equalization

CLAHE (Contrast Limited Adaptive Histogram Equalization) is an image processing technique used to improve the contrast and visibility of images, particularly in regions with low contrast or poor lighting conditions.

In a traditional histogram equalization, the dynamic range of the image is spread out to cover the full range of possible pixel values. However, this can lead to the over-amplification of noise in areas with low contrast or inhomogeneous illumination. CLAHE addresses this problem by dividing the image into small regions and applying histogram equalization to each region separately while limiting the amplification of noise. This ensures that contrast enhancement is adaptive and localized, rather than globally applied to the entire image.

In CLAHE, the contrast amplification in each region is controlled by a contrast limiting parameter, which specifies the maximum amplification factor allowed. This helps to prevent over-amplification of noise, while still improving the contrast and visibility of the image. The resulting image has improved detail and texture, making it easier to visualize and analyze.

CLAHE has applications in various fields, including medical imaging, remote sensing, and computer vision. It is commonly used as a preprocessing step before applying other image-processing techniques, such as segmentation or feature extraction.

Symmetric Check in Vitiligo Research

Vitiligo is mostly symmetrical which lead to the idea to split the image at the nose line and then flip the left side and match it over the right side.

The idea came because if we look at the image e.g. a face normally the left and right sides from the nose line are equal. This means the skin on the left Figure 13A) and right side Figure 13B) and the XOR result in Figure 13C)

has the same skin temperature. When we look at vitiligo some patients have different patterns on the left and the right side. If we split the image on the nose line which we can get via the face location and face mask. We can resize the left side to match the right side. With the sides aligned, we can flip the left side and overlay both images.

Symmetry check is a useful approach for comparing the sides of a patient image in vitiligo detection. Vitiligo often affects both sides of the body symmetrically, which means that corresponding areas on the left and right sides of the body should have similar shapes, sizes, and pigmentation patterns. By comparing the symmetry of the affected areas on both sides of the body, it is possible to detect and quantify the extent and severity of vitiligo.

To perform a symmetry check in vitiligo detection, the patient image can be split into two halves along the midline, and each half can be compared to the other using image processing techniques. For example, the intensity values or color values of corresponding pixels on both halves can be compared, and a different image can be generated to highlight any areas of asymmetry. Alternatively, geometric features such as shape, size, and position can be extracted from the affected areas and compared between the two halves using mathematical or statistical techniques.

Symmetry checks can be used as a complementary approach to other vitiligo detection methods, such as color thresholding and edge detection, to provide a more comprehensive and accurate analysis of the patient image. However, it is important

to note that some degree of natural asymmetry is normal in human bodies, and not all cases of vitiligo may exhibit perfectly symmetric patterns. Therefore, symmetry check should be used as a guide and not as the sole criterion for vitiligo diagnosis or treatment. If the skin on one side is different than on the other side this could be an indication of Vitiligo.

Figure 13) Symmetry Check

Histogram Matching in Vitiligo Research

Histogram matching is a technique used in image processing to transform the pixel values of an image so that they match the histogram of a reference image. The goal of histogram matching is to equalize the pixel intensities between the two images, which can be useful for tasks such as image registration and color correction.

Here's a rough outline of how histogram matching works:

1. Collect two images, the source image and the reference image.

2. Compute the histograms of the source and reference images.

3. Compute the cumulative distribution functions (CDFs) of the source and reference histograms.

4. Compute the mapping function that maps the pixel intensities of the source image to the pixel intensities of the reference image. This can be done by finding the inverse of the CDF of the reference image and applying it to the CDF of the source image.

5. Apply the mapping function to the pixel values of the source image to obtain the matched image.

The basic idea behind histogram matching is to make the pixel values of the source image more closely resemble those of the reference image. This can be useful for a variety of applications, such as color correction in photography, or in medical imaging to match the intensity levels of different scans of the same patient.

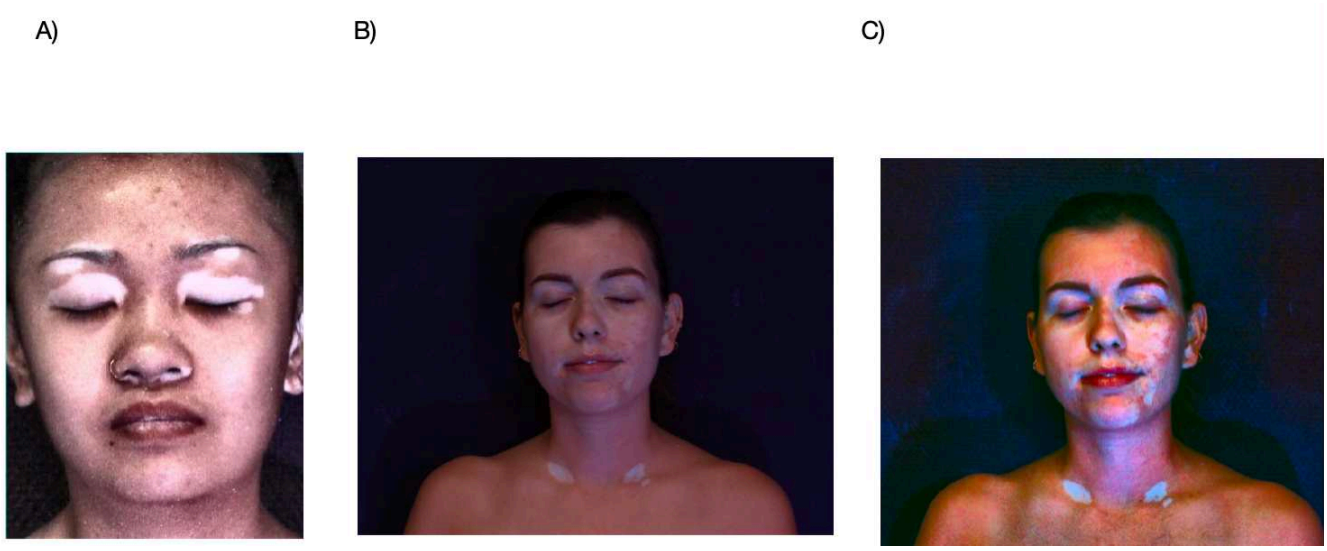
The problem with different lighting conditions in images can be addressed using histogram matching. Histogram matching allows us to adjust the pixel intensities in an input image to match the pixel intensities' distribution in a second reference image. This alignment can help to ensure that the images have the same appearance and contrast, even when they were captured under different lighting conditions.

In the case of vitiligo detection, different lighting settings can affect the visibility and contrast of the affected skin patches, making it challenging to compare images. By normalizing the images with a suitable reference image, we can align all the images to a common histogram and threshold that can be used consistently across all images. However, finding the ideal matching image can be challenging and may require trial and error.

Therefore, while histogram matching can be a useful technique for addressing differences in lighting conditions, it is not a complete solution for detecting vitiligo. In Figure 14A) we have a selected template. From this template the histogram information is transferred to image 14B) and the result is the histogram-aligned image in Figure 14C). Other techniques such as segmentation and feature extraction

may also be required to accurately identify and quantify the extent of the affected skin patches.

Figure 14) Histogram Matching



A) Template Image. B) Original Image C) Equalised Result image

Histogram matching can best be thought of as a “transformation.” While the input image’s actual contents do not change, the pixel distribution does, thereby adjusting the illumination and contrast of the input image based on the distribution of the reference image.

Histogram equalization is typically performed on a single image to enhance its contrast. If we want to match the color distribution of one image to another (reference) image, we can use histogram matching or color transfer techniques.

Color transfer is a technique used to harmonize the color distribution and appearance between two images. It aims to make the colors of one image more similar to the colors of another image while preserving the content and structure of the original images.

The basic idea behind color transfer is to align the color statistics (such as mean, variance, and higher-order statistics) of the target image with those of the source image [52][51]. This can be achieved through various methods, and here is an outline of a common approach using color histograms:

1. **Convert Images to a Common Color Space**: Start by converting both the source and target images to a common color space. Common choices include RGB, Lab, and HSV color spaces.

2. **Compute Color Histograms**: Calculate color histograms for both the source and target images in the chosen color space. Color histograms represent the frequency distribution of colors in an image.

3. **Normalize Histograms**: Normalize the histograms to account for differences in image size.

4. **Histogram Matching**: Adjust the color distribution of the target image's histogram to match that of the source image's histogram. This can be done by manipulating the cumulative distribution functions (CDFs) of the histograms.

5. **Apply Color Mapping**: For each pixel in the target image, find the corresponding color value from the source image using the histogram mapping obtained in the previous step.

6. **Apply the Transferred Colors**: Replace the colors of the target image with the transferred colors.

7. **Optional Post-Processing**: Some methods include post-processing steps to fine-tune the color transfer result, such as gamma correction or additional smoothing.

Color transfer techniques can range from simple histogram-based methods to more complex algorithms that take into account local color relationships or consider spatial information. Some methods use statistical measures like the covariance matrix to capture color correlations more accurately.

It's important to note that while color transfer can achieve good results in many cases, it might not always produce the desired outcomes, especially when images have vastly different content or lighting conditions. Implementations of color transfer can be found in various image-processing libraries, and custom implementations can be created using tools like Python and OpenCV.

Inverse Method in Vitiligo Research

To improve the detection of vitiligo in images, one approach is to invert and darken the affected area, followed by blurring to enhance visibility and ease of locating. This technique helps in setting better parameters for the vitiligo detection algorithm. By

making the vitiligo area darker and applying a boundary, the tracking process becomes easier, resulting in more accurate localization.

Manual Creation of Masks

The creation of masks through the annotation of images is a labor-intensive task, often considered a last resort when automated methods are not available. In our study, this process became necessary due to the presence of white hair or beard sections that shared the same color as non-pigmented skin areas in some patient images. But also some skin tones did not allow an automatic creation the non-pigmented skin was lying in an area that covered already different pigmented skin regions. A segmentation of the image into four parts and separate processing to reduce the complexity did also not solve the issues encountered.

To address this challenge, we manually annotated images by marking them in such a way that image and mask patches could be generated. Specifically, we annotated 700 patches corresponding to hair and beard regions in the images. These annotated images were then employed as part of our data preparation for training a UNet network.

Patient report in Vitiligo Research

To streamline the process and enhance patient experience, the idea of creating an online portal for patients has emerged. This portal will serve as a platform to collect and organize patient information, including images and corresponding comments, in a structured manner. The patient report should contain information about the total amount of vitiligo in the body that was detected at the time. In general, the idea was to present the original image the calculated mask, and as a result the overlay of the original image and mask with an additional canny overlay to make the non-pigmented skin areas more prominent.

PDF Reports

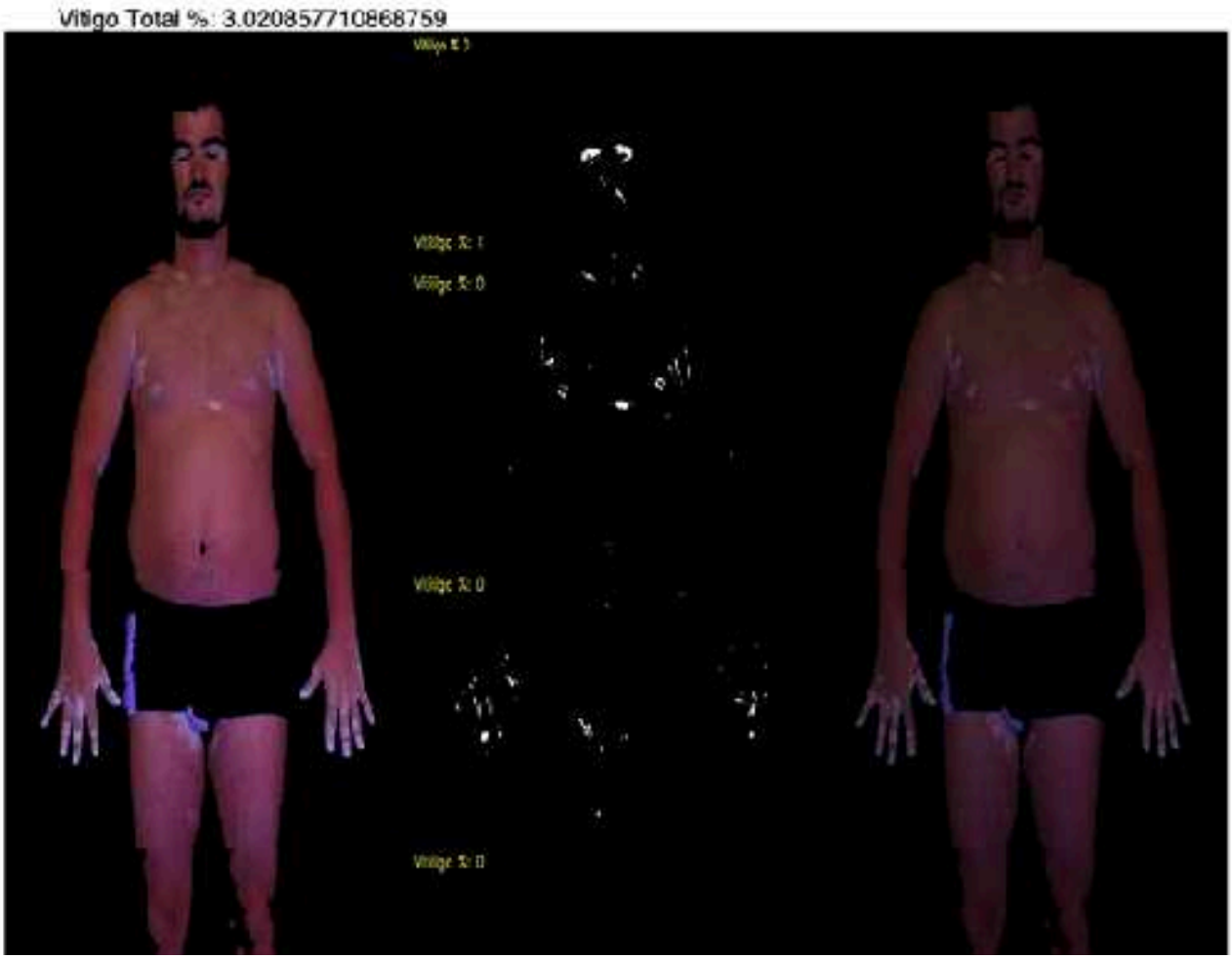
The patient reports will be generated and presented within the portal. Each picture taken during the examination will be allocated its dedicated section within the report. The report will also provide an option to include additional comments related to each specific image. To facilitate easy data interpretation, an automated table format will be generated, allowing for seamless insertion of results. We will produce a front and back report as well as a left and right side report. The remaining images are added in a loose format since we can not concatenate them.

To ensure convenient accessibility and sharing of patient reports, the finalized documents will be generated in PDF format. However, it is worth noting that the placement and arrangement of images within the PDF document may pose some

challenges due to formatting complexities. Figure 15) shows a sample. Conctatnated images of the original images are presented on the left Figure 15A) the

Figure 15) PDF Report

A) B) C)



A) Concated Orignal Images B)Generated Mask. C) Overlay of Original and Mask

vitiligo mask is presented in the middle Figure 15B) and the result an overlay of the mask and a canny algorithm is presented on the right Figure 15C).

By implementing the patient online portal, we aim to enhance the efficiency and effectiveness of the patient-care process, promoting better communication, documentation, and data organization. The portal will provide a centralized hub for

all relevant information, facilitating collaboration among healthcare professionals and ensuring a comprehensive overview of each patient's case.

A different approach was to take the original image concatenate the calculated mask image and then concatenate the original image with the overlay of the original and contours of the mask. This will provide a good overview of the current and the calculated and wrongly selected areas of vitiligo that can easily be spotted. Other possibilities like online access are also possible and discussed in the following section.

The alternative is to create a presentation of the original image as a reference, the generated mask, and an overlay as a result of the overlay's original image by edges of the mask generated and some parts of the mask to make the vitiligo region more present.

Powerpoint integration

The results obtained from the image processing are exported as images to the file system. However, to provide a more suitable and organized presentation of the results, a demand arose for a more appropriate format. In response to this, a solution was developed to generate PowerPoint slides for each processed image. The label on each slide is set to the corresponding filename, serving as a reference, and includes the calculated percentage of vitiligo. This approach enables the creation of a comprehensive presentation that can include multiple images, facilitating the interpretation and diagnosis of patient information by clinical personnel.

To implement this functionality, Python libraries for creating PowerPoint slides were utilized. By incorporating this feature into the vitiligo processing algorithm, the generation of PowerPoint slides becomes an integrated part of the workflow, streamlining the presentation of results and enhancing their accessibility for clinical evaluation.

This approach offers a practical and efficient means of visualizing and sharing the vitiligo analysis outcomes, ensuring that the relevant information is presented clearly and concisely.

CSV Reports

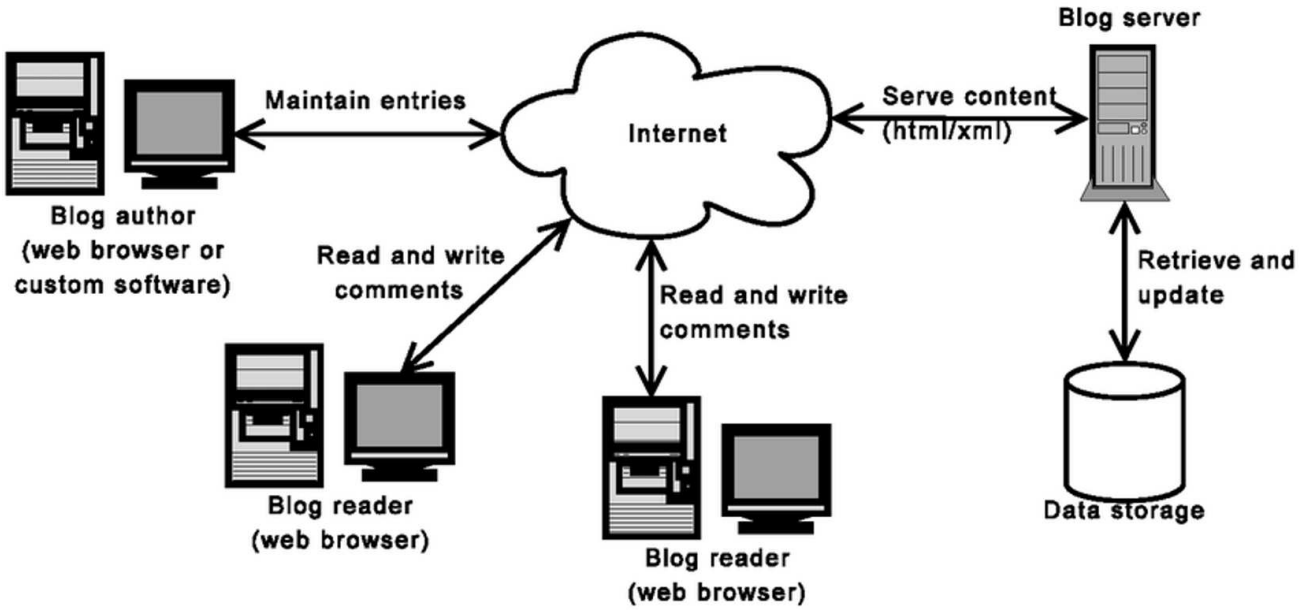
The processed images are not only presented visually but are also reported in a structured and easily accessible format. To enable further analysis and tracking of patient progress, the results are exported in a CSV (Comma-Separated Values) format. This CSV file contains essential information such as the filename, vitiligo percentage, and corresponding settings used during the analysis.

By having the data in a CSV format, it becomes convenient to import and analyze the results in various data processing tools or software. This allows for efficient comparison and tracking of improvements in patients over time. The structured layout of the CSV file ensures that the information is easily readable and can be used for further statistical analysis or data manipulation as needed.

Creation of an Online Platform for Vitiligo image processing.

An online platform for Vitiligo image processing can bring several benefits to both healthcare providers and patients Figure 16).

Figure 16) Online System Architecture



Online System Architecture

About the distance, some patients need to overcome the time and effort it takes to visit medicine centers, and online systems have started to become more relevant than before. Some reasons why an online platform can be useful:

1. Improved accessibility: An online platform can allow patients to access medical services remotely from anywhere at any time. This can be especially helpful for patients who cannot visit a physical clinic due to distance, disability, or other reasons.

2. Convenience: Patients can easily book appointments online, and healthcare providers can manage their schedules and patient records from a centralized platform. This can save time and reduce the need for manual processes.

3. Better patient engagement: An online platform can provide patients with access to their medical records, test results, and other health-related information. Patients can also receive alerts, reminders, and educational resources that can improve their engagement with their healthcare providers.

4. Increased efficiency: Online platforms can enable healthcare providers to streamline their workflows, reduce paperwork, and automate routine tasks. This can free up time for providers to focus on more critical aspects of patient care.

5. Cost savings: An online platform can reduce the need for physical infrastructure and staffing, which can result in cost savings for healthcare providers. It can also reduce healthcare costs for patients by providing access to preventive care and early interventions.

Overall, an online platform for a Vitiligo image processing service can improve the quality of care, increase patient satisfaction, and reduce healthcare costs.

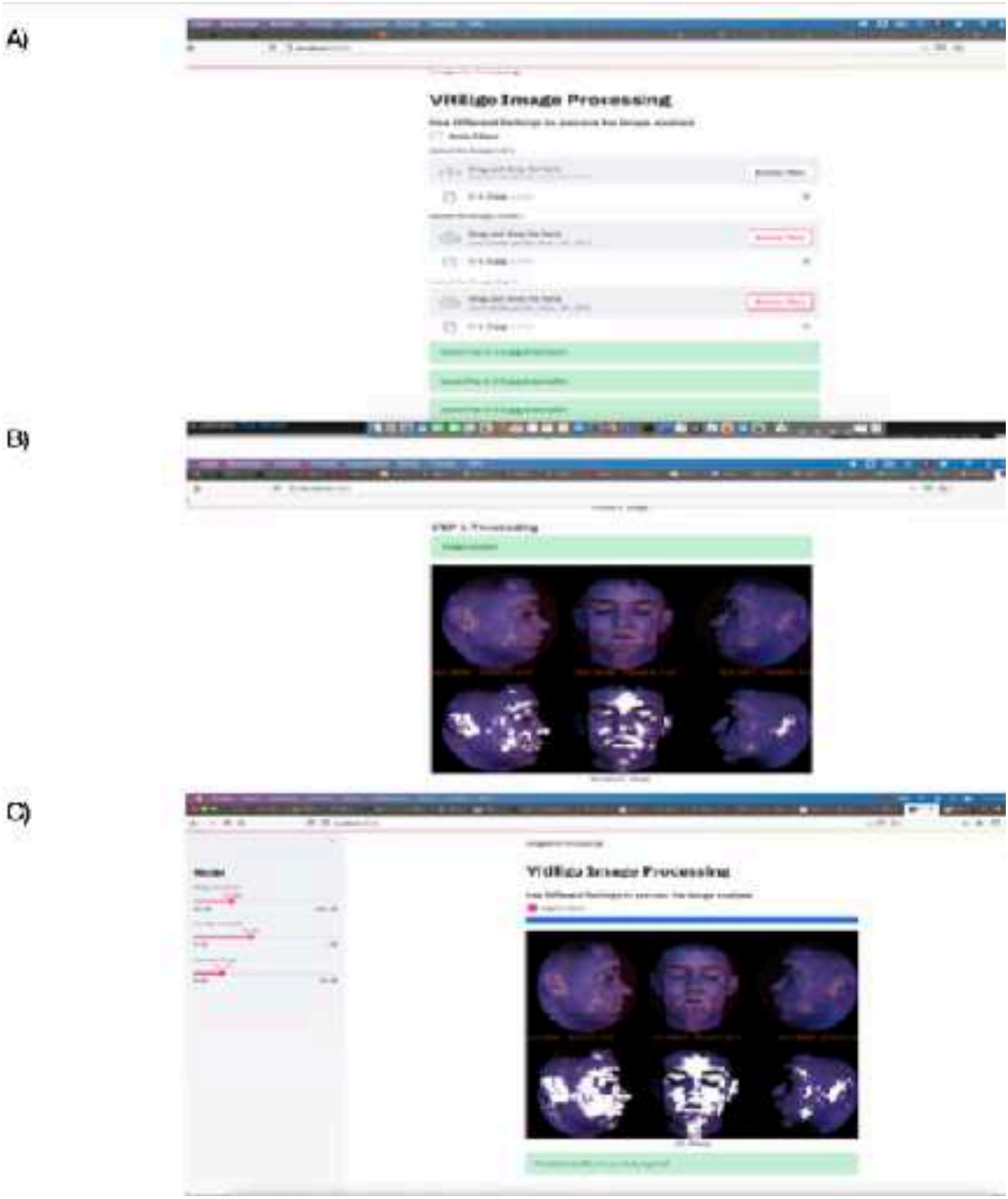
The first version of an online platform was created with a web server. The implementation allows the upload of images on a website and then processing the images and presenting the results. An online platform can be designed as a blog-like space for patients to create their entries and upload data and images. Access to this data can be restricted to the creator of the entry or authorized administrators (in this case, physicians who are examining patient information).

The Flask software project, which is built on Python, can be used to implement this platform. This network provides a robust framework for the required functionality. The patient data and access information can be stored in a MySQL database, while the images can be stored with encoded file names in the file system. This approach helps to anonymize the image data and prevent unauthorized access to patient information.

The image’s two sides and the front of the face can be uploaded. The next step is to create a vitiligo calculation (shown in Figure 17A). The images will be processed and the result be presented on the website as a return value this is shown in Figure 17B). The resulting image is available and can be reviewed some parameters can be changed a slider is available to adjust the thresholds this is shown in Figure 17C). The amount of slider that would be required to adjust the images separately would be counterproductive in the clinical process. Therefore this approach was stopped.

Overall, this approach can be a useful way to facilitate communication and data sharing between patients and healthcare providers, while also protecting patient privacy and ensuring data security. However, it is important to ensure that appropriate security measures are in place to prevent data breaches or unauthorized access to sensitive patient information.

Figure 17) Online User Interface



A) Upload of Patient Images. B) Image Processed on Server C) Image Adaption Page

Flask

Flask is a popular web framework for Python that allows developers to quickly build web applications. It is a lightweight and flexible framework that is often used for small to medium-sized projects. Flask provides developers with the tools needed to create a web application, including a web server, routing system, and template engine. It is known for its simplicity, ease of use, and extensive documentation. Flask is also modular, meaning that developers can use only the parts of the framework they need for their specific application, making it highly customizable. Additionally, Flask supports a wide range of third-party extensions and libraries, allowing developers to add functionality to their applications easily.

A flask is a web server that was extended or tailored for specific needs:

Running the Image Processing
Storing in Database
Access for the Patient
Access for the Physicians
Images Stored in Database
and File System

The server has a small footprint and is easy to configure. The provided libraries and samples such as a Blog can easily be modified to fulfill the needs of a patient administration system as we tend to do.

Flask as software is based on the **W**eb **S**erver **G**ateway **I**nterface toolkit and Jinja2 template engine. In a traditional web server, there is no possibility of running Python applications. The Apache module was created for executing Python codes. This module was not a standard and therefore lacked some security which led to open doors for vulnerabilities which caused the Python community to take action create a WSGI interface, and create the Flask server. This enriched the possibilities about the following:

- **Flexibility**: The modules used can be decided as needed which allows to creation of small applications running on limited resources. Allowing the development of tailored software on small devices.

- **Scaling**: It is possible to scale the servers so that multiple connections can be established at once. This allows the connection of multiple clients sending requests to be handled.

Flask has a built-in server component. The best is to use Flask in combination with a web server for better performance. This allows then the use of different options. On

one side the communication and also the session handling. These are some key aspects concerning web servers. The benefits are:

- **Handling static files:** Web Applications use static files for the rendering of websites such as CSS etc. The web server can be enriched with the use of Flask to present content.
- **Flask sessions**: The sessions are handled so the session management is encapsulated and can be utilized in the Flask application.
- **Uploading of File in Flask**: The Web server buffers the information first before they are sent to the Flask application which allows better data handling.
- **Sending Form data:** The page data containing the HTML code is built and then sent via the Web server to the clients.

The patient data and the images fit perfectly with the above-mentioned facts. The use of a MySql database for secure storage and the use of the filesystem for image data make the Flask server implementation a close fit. The database schema we created contains the user information and a link to the file names. The filenames are created by use of a unique number which is via the database table referencing the user information. With the filename itself, the patient can not be located in the filesystem.

The User interface allows new persons to be registered and existing users to log in. When a user is logged into the system he can see his data and add new data as shown in Figure 18A). He can upload new images. The deletion of data is not allowed. The image is processed in the server in the background and a result will be presented as another post as shown in Figure 18B)

For the physicians the information of all patients is visible. The patent data can be reviewed and commented on. The data can not be deleted. The connection is done via the Internet. The Mobile Phone Application or the Web Browser can be used to upload information. Available for the Flask server are a variety of applications on a BLOG that are quite useful to be used as a patient system. The information can be made visible to only the author and the administrators can read all since they are treated as administrators.

Flask uses routes and we can since we use only a small amount of the functionality create a quite flexible application.

The routes describe the communication that should be created. We have a login screen and we have a post screen to enter the data.

Another functionality is the possibility to provide an upload for image or video data. Since a Blog is normally Text and images that is already a satisfied requirement for our application.
If a user logs in to the system it will present the main screen that will show a list of all entries made from him.

With the use of MySql an open-source Database Management System we can create our data schema. The basic schema contains 2 tables one for the User information and one for the data references.

The Flask allows us easily to connect to a database. It is possible to create our database schema. The information about the user and the information about the data like images need to be stored in a database. This allows us to keep the information safe. The database schema holds information about the user like password and address information. The information like images etc are stored in the file system and will use a GUID that is only in the database related to the user record. Having access to the filesystem and looking at the images they will lack the information to which patient they are belonging.

The Flask system is also based on Python. The procession algorithm for vitiligo calculations can be easily incorporated. Flask allows the execution of Python programs. The result is a result image containing three images original, a vitiligo mask, and an overlay image of the original and the contours of the vitiligo mask. The image is saved in the filesystem and then the generated GUID is used as a filename stored in the database as a result value for the execution.

The information is then presented to the patient and the physician to be reviewed. Comments and information can be shared on the platform as well. The hoovering over the resulting image will open it in a larger window which allows a simpler analysis of the image.

Mobile Solution

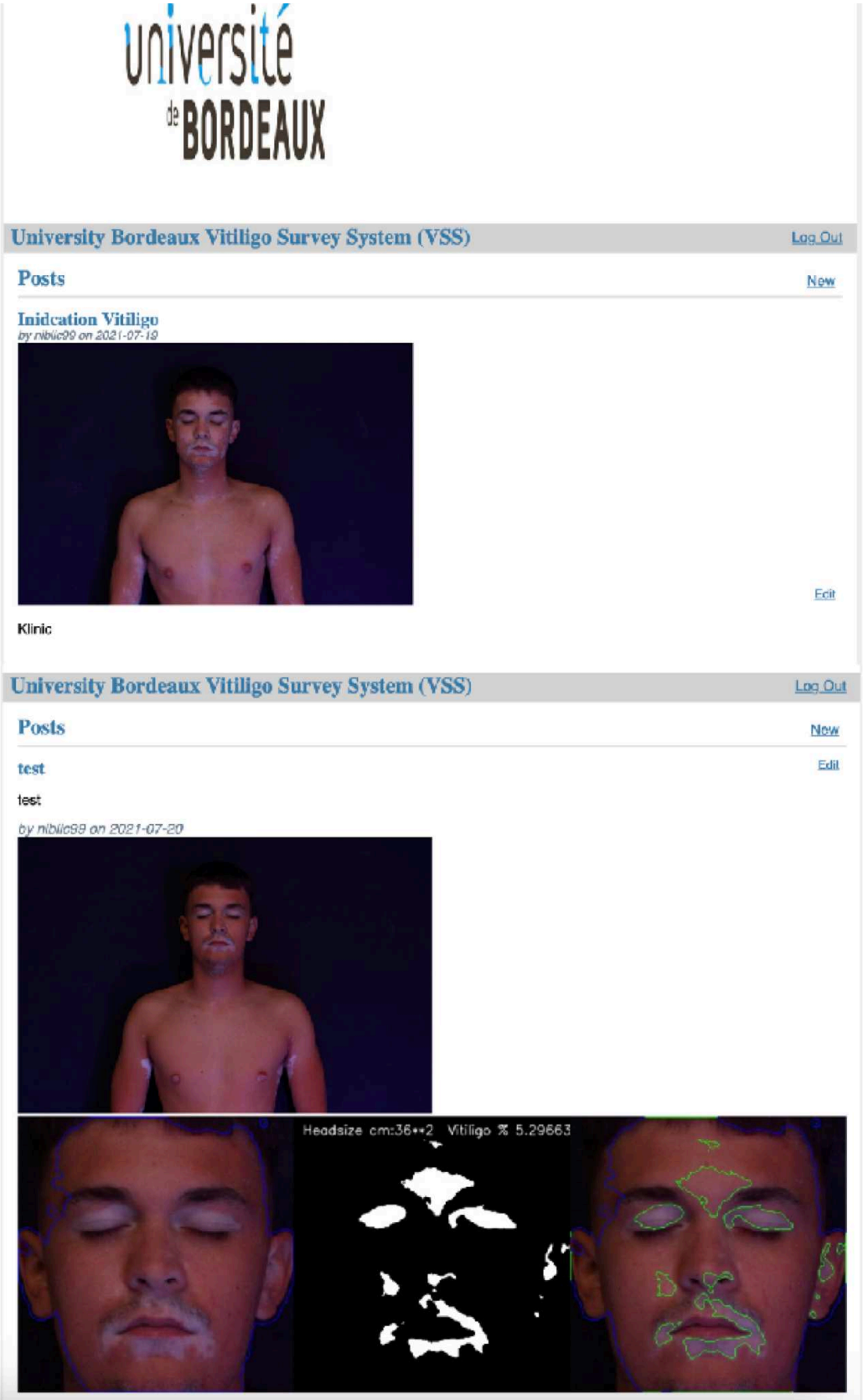
Medical consultancy is not in every country as easy as in France or Germany etc. some countries are less well equipped or medical personnel is hard to reach. Telemedicine is one possibility to respond to the need for medication also in remote areas. In Australia and other countries like India, the computer is used to make video conferences and conduct consultations.

This led to the idea of providing a possibility for patients to contact a physician via mobile device a phone or tablet or a laptop. The idea is to create a simple app that allows the creation of images and uploads to a central server. The mobile solution

was created in an Internship by Anthelme Pradeau for the University of Bordeaux. The mobile networks are strong enough to transfer images. The focus is on the head since this can also be taken without assistance. After the transfer of the images to

Figure 18) Flask Online Interface

B)

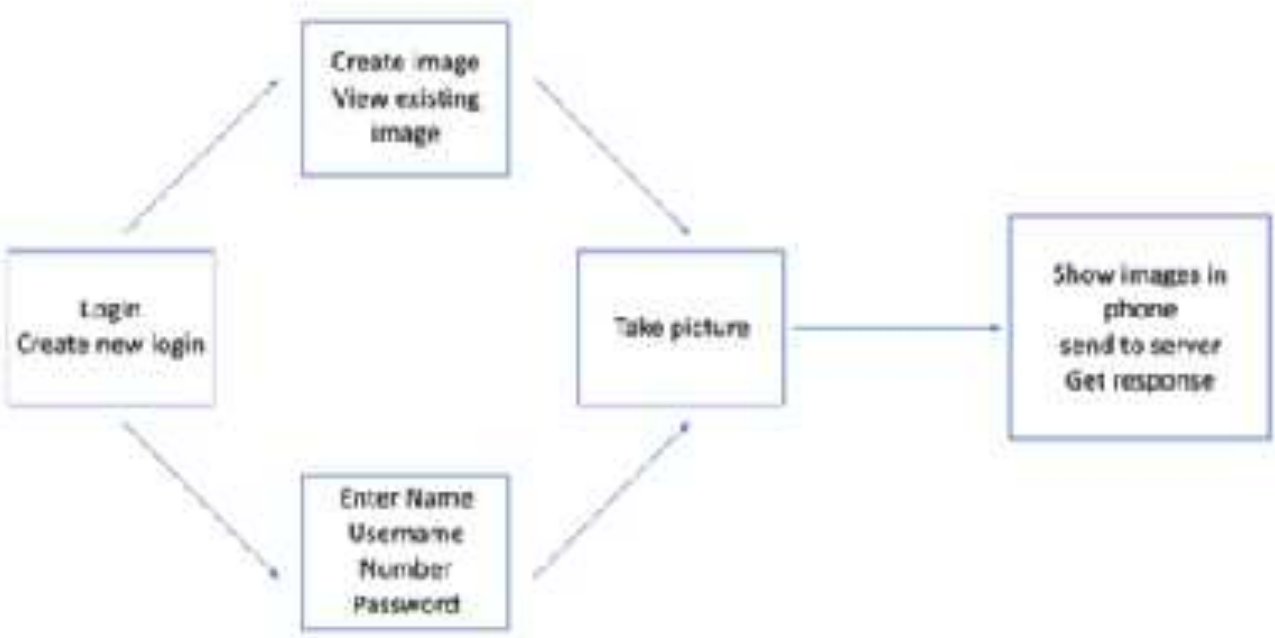


Caption

the server the images can be checked and a health report can be sent back to the architecture shown in Figure 19). The UI provided the following flow:

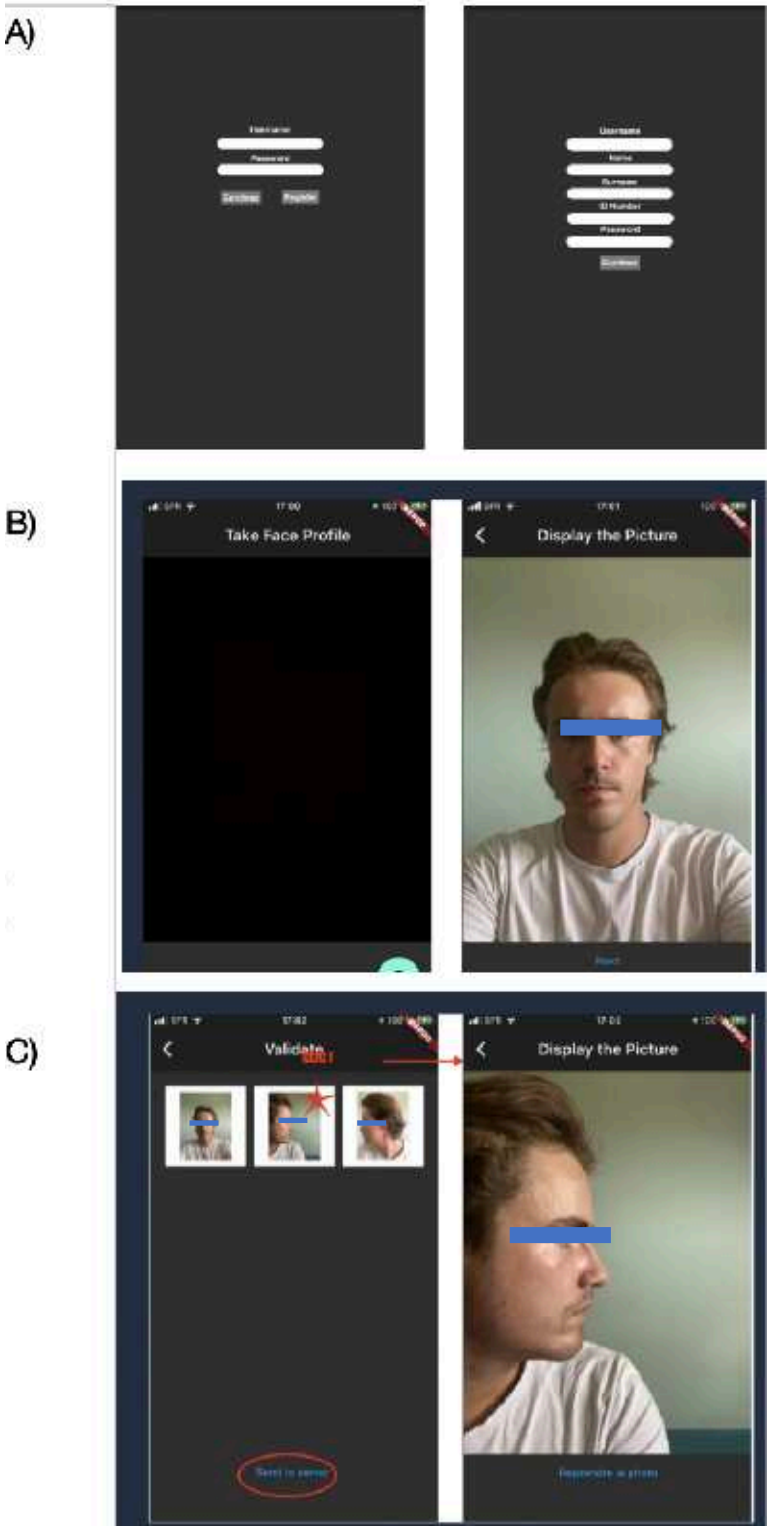
1) Login or create a new account
2) Create an image for analysis
3) Enter the Username and Password for the existing account
4) Take picture
5) Store image in phone and send to server get response and display results

Figure 19) Mobile Data Flow



The implementation consists of a Mobile application that will handle the login to the backend system a Flask Server. Figure 20A) shows the login screen for an existing and a new user.

Figure 20) Mobile User Interface



Caption

A) Login and User Registration B) Adding new Images C) Collecting Images for Transfer

Figure 20B) allows the user to take his picture. In Figure 20C) the images are collected and can then be transferred to the server. The server is connected to a database that contains the user ID as well as passwords. The images uploaded will be saved in the filesystem with a GUID as file name and a reference which is stored in the database and connected with the user record. Access to the system allows the patient to see and comment on his images or records. The physician has access to his patients and can view and edit all patient data.

Images taken can be checked by the patient before being transferred to the server. This reduces the amount of data that needs to be transferred. Only the good quality with a minimum of skin reflection would be a good resource for the vitiligo analysis. If there is too much reflection on the skin the algorithms can lack accuracy.

Processing on server

In the mobile solution, the information collected from the mobile phone application is transmitted to the Flask server for further processing. Once received, the data is processed and stored in both the database and the filesystem. The Flask server then invokes the Vitiligo program, which performs the necessary image-processing tasks. The output of the program is an image that depicts the vitiligo percentage, and this image is stored in the filesystem. Additionally, the server updates the database with the filename information of the generated image.

It is worth noting that the image processing on the server is particularly challenging when working with images captured under natural lighting conditions. The location of vitiligo patches becomes more difficult to discern, as the absence of UV light, which is commonly used in clinical settings, reduces the visibility of vitiligo. In cases where the vitiligo is confined to specific areas, such as around the eyes, it becomes even more challenging to accurately analyze and detect the affected regions. With natural light, the analysis requires careful examination and the trained eye to identify subtle signs of vitiligo that may not be immediately apparent.

The findings suggest that the use of UV light is crucial for effective vitiligo analysis, as it provides enhanced information and visibility of the vitiligo patches. Further research and experimentation may be necessary to develop improved image processing techniques that can better handle natural light conditions and accurately detect vitiligo, particularly in cases where the affected areas are limited and subtle.

AI Networks in Vitiligo Research

Artificial neural networks (ANNs), also known as AI networks, are computational models inspired by biological neural networks in the human brain. They have

become a fundamental tool in machine learning, with applications ranging from image recognition and natural language processing to speech recognition and robotics.

Various types of AI networks exist, each with its characteristics and applications. Here are some common types:

1. Feedforward Neural Networks: These networks process information in a single direction, from input to output. They are widely used for tasks such as image recognition and classification.
2. Recurrent Neural Networks: With loops in their architecture, recurrent neural networks excel at processing sequential data, including time series, speech, and natural language.
3. Convolutional Neural Networks: Designed for grid-like data, such as images and videos, convolutional neural networks are essential in tasks like object detection, segmentation, and classification.
4. Generative Adversarial Networks: These networks involve a generator and a discriminator network that compete against each other, resulting in the generation of new data that resembles the training data. They are useful for generating images, text, and other synthetic data.
5. Autoencoder Neural Networks: Autoencoders learn to encode and decode data in an unsupervised manner, enabling tasks like data compression, feature extraction, and anomaly detection.

For vitiligo detection, the Convolutional Neural Networks are suited the best. The UNet architecture allows the processing and matching of patterns in a very efficient way with high accuracy.

When working with AI networks, obtaining sufficient training data is crucial. Manual data annotation can be time-consuming, necessitating automated approaches to improve the process. AI networks typically require large amounts of data to achieve satisfactory results. An accuracy of 50% would be akin to random guessing and not a reliable decision.

While the amount of data required varies among different network types, generating and training networks with limited data has been attempted with varying degrees of success. Automation of data generation can help address these challenges and improve the efficiency of training processes.

CNN Sequential

A CNN Sequential network is a convolutional neural network (CNN) architecture that consists of multiple layers arranged in a sequential order. This architecture is

commonly used for image recognition and classification tasks. The network begins with convolutional layers that apply filters to the input image, extracting high-level features. These features are then passed through pooling layers to reduce dimensionality and preserve important information. After several rounds of convolution and pooling, the feature maps are flattened and fed into fully connected layers for classification or regression tasks. The sequential nature of the network allows it to learn increasingly complex representations of the input data.

The CNN Sequential network requires test and validation data to assess its performance. The choice of activation function is important for network training. We experimented with both the Rectified Linear Unit (ReLU) and Sigmoid activation functions.

ReLU Activation: ReLU tends to saturate or close up early in the training process, which can limit its effectiveness. This is a known issue with the ReLU activation function. To address this problem, alternative activation functions like Sigmoid or others can be considered. The CNN Sequential network was trained in a short amount of time, but reevaluating the results by reloading the dataset may be necessary.

Sigmoid Activation: Sigmoid is a more robust activation function that provides better training results. It ensures that the output values are between 0 and 1, allowing for more equalized information flow within the network. Using the Sigmoid activation function can enhance the training process.

This led to the development and implementation of a more specific network architecture known as the UNet. The UNet is a convolutional neural network that has demonstrated excellent performance in image segmentation tasks, including vitiligo segmentation.

UNet Comparison in Vitiligo Research

Vitiligo segmentation plays a crucial role in evaluating the effects of treatment on patients' skin. Currently, this evaluation process relies heavily on manual assessment by experienced physicians. By introducing AI techniques, this task can be automated and standardized, allowing for better comparison with other images or future assessments.

In our research, we utilized an Attention Resilient UNet implementation [62], as described in [39]. The fundamental concept behind UNet involves downsampling during the encoding phase and upsampling during the decoding phase. The standard UNet employs a simple cost function and provides basic functionality. To

enhance the performance of our model, we used the dice coefficient [40] to calculate the intersection of matched data. This technique involves determining the intersection of different mask patches and assessing their occurrence.

Accuracy alone is not a vital indicator for the network in this context. Since the patches are defined with both skin and vitiligo regions, we consider the background data, representing normal skin, to be less significant. Consequently, the accuracy metric tends to emphasize the background. However, given the large amount of background data, the accuracy metric initially produces relatively high values.
To address the segmentation task more accurately, we implemented the Jaccard coefficient.

$$J(A,B) = \frac{|A \cup B|}{|A \cap B|}$$

The Jaccard coefficient calculates the equality of unions between two segments, enabling a more precise calculation of the loss function. This helps overcome issues related to local minima and generally leads to better results.

The Dice coefficient is calculated as follows:

$$Dice(A,B) = \frac{(2 * |A \cap B|)}{(|A| + |B|)}$$

In the formula:

- A represents the ground truth (the manually annotated or true segmentation mask).
- B represents the predicted segmentation mask generated by your UNet model.
- |A ∩ B| denotes the intersection of regions between the ground truth and predicted masks.
- |A| represents the total number of pixels or elements in the ground truth mask.
- |B| represents the total number of pixels or elements in the predicted mask.

This formula calculates the overlap between the predicted and ground truth masks, providing a measure of segmentation accuracy where 1.0 indicates a perfect match and 0.0 indicates no overlap.

The UNet architecture is a popular choice for semantic segmentation tasks [41], such as medical image segmentation, where the objective is to assign each pixel in an image to its corresponding class or category. Introduced in [47] it has since become a widely used convolutional neural network (CNN) architecture.

The UNet follows an encoder-decoder structure with skip connections[42]. The encoder part captures the contextual and spatial information of the input image through convolutional and pooling layers, progressively reducing its spatial dimensions. On the other hand, the decoder part employs upsampling and convolutional layers to reconstruct the segmented output, leveraging the skip connections to combine low-level and high-level features[43]. This integration allows for precise localization and effective segmentation.

By leveraging the UNet model, we can achieve accurate vitiligo segmentation in input images. The network is trained using a combination of image patches and corresponding mask patches, which serve as the ground truth for segmentation[44]. Through an iterative training process, the UNet learns to accurately classify pixels within the image and differentiate between vitiligo and non-vitiligo regions.

The UNet has demonstrated remarkable performance in vitiligo segmentation, exhibiting impressive accuracy and reliability. However, it is important to note that the success of the UNet heavily relies on the availability of diverse and representative training data[45]. Particularly for vitiligo segmentation[46], it is crucial to have training data that encompasses various skin colors, temperatures, and lighting conditions to ensure optimal results.

In conclusion, the UNet architecture has emerged as a valuable solution for vitiligo segmentation, offering a robust and effective approach to identifying and differentiating vitiligo regions within images. Its incorporation of skip connections and its ability to leverage training data for accurate pixel-level classification make it a powerful tool in medical image analysis.

There are some variations of the UNet network available the main differences are the back flow of information and the creation of intersections that show the important areas.

We use the Attention Residual UNet for vitiligo recognition. The Attention Residual UNet is a UNet with changed processing. The upsampling will use an Attention gate:

1. The attention gate takes in two inputs, vectors x and g.

2. The vector, *g*, is taken from the next lowest layer of the network. The vector has smaller dimensions and better feature representation, given that it comes from deeper into the network.
3. In the example figure above, vector ***x*** would have dimensions of 64x64x64 (filters x height x width) and vector ***g*** would be 32x32x32.
4. Vector ***x*** goes through a stridded convolution such that its dimensions become 64x32x32 and vector ***g*** goes through a 1x1 convolution such that its dimensions become 64x32x32.
5. The two vectors are summed element-wise. This process results in **aligned weights becoming larger** while unaligned weights become relatively smaller.
6. The resultant vector goes through a ReLU activation layer and a 1x1 convolution that collapses the dimensions to 1x32x32.
7. This vector goes through a sigmoid layer which scales the vector between the range [0,1], producing the attention coefficients (weights), where coefficients closer to 1 indicate more relevant features.
8. The attention coefficients are upsampled to the original dimensions (64x64) of the ***x*** vector using trilinear interpolation. The attention coefficients are multiplied element-wise to the **original *x*** vector, scaling the vector according to relevance. This is then passed along in the skip connection as normal.

The Performance of the network depends on the training data. But not only the quality means exact masks that are provided for training also the amount and frequency of training are important and make a difference. The assurance of the network is not a real indication of the correctness of the predictions. It is just an indication the use of new data is then proof.

Creation of Training Data for Head Images in Vitiligo Research

The image is processed using an HSV-converted image, and the image channels are separately examined to set high and low boundaries. The algorithm proceeds as follows:

1. Locate the skin in the image to remove the background. Although not all background noise can be eliminated, efforts are made to minimize it.
2. Identify the face in the image. If analyzing other body parts is required, this step can be skipped.
3. Determine the minimum V value and assign it to min_v.
4. Determine the minimum S value and assign it to min_s. Assume the maximum S value to be 42 values higher than min_s.
5. Find the maximum H value and assign it to max_h.
6. Create an output mask using the following settings:
    - Low H, S, V values: (0, min_s, min_v)

    - High H, S, V values: (max_h, max_s, 252) The values for min_h and max_v can be set to default values as they have minimal impact on the color selection in question.
7. The generated mask, along with the extracted head image or normal image, can now be processed to generate CNN training and test data.
8. The information from the mask is used to determine whether an image patch should be classified as Vitiligo or normal skin. The current patch size is set to 128x128 but can be adjusted if needed.
9. Image patches are saved in different orientations.
10. The information is then organized and stored in separate folders.
11. The CNN test and training data will utilize the folders for classification definitions.
12. The training of the network and validation of the trained network are defined during the data loading process.

By leveraging the knowledge gained from the image mask creation process, we discovered that the HSV color scheme enables the localization of non-pigmented skin by adjusting the S channel. Specifically, we identified the minimum S value for cases where the image did not contain any white spots, such as white or grey hair or jewelry. With this information, we were able to establish an upper boundary for S (S max) by using the AI-generated mask as a reference. We increased the S max value iteratively and evaluated if the resulting mask contained all the relevant pixels from the AI mask. Consequently, the AI mask served as a valid indicator for determining the upper boundary, facilitating the creation of more accurate vitiligo masks.

A similar approach was applied to adjust the H min, H max, and V max values. By using incomplete masks, we extended this technique to patches, enabling a more detailed analysis due to varying lighting conditions. For patches with an AI mask indicating non-pigmented skin, the same process was applied, adjusting the S max, H min, H max, and V max values. This approach facilitated the identification of previously undiscovered non-pigmented areas.

One advantage of this approach is that it allows for a more granular examination of the image by analyzing patches instead of the entire image. However, it comes with the drawback of potentially capturing.

The effectiveness of the U-net network in processing AI outcomes is intricately linked to the quality and quantity of training data. As a supervised learning model, the U-net network relies on annotated data to accurately segment images. This annotated data consists of pairs of input images and corresponding labeled masks, where each pixel in the mask is assigned a label indicating its class.

Insufficient or low-quality training data can impede the U-net network's ability to learn relevant features and patterns necessary for accurate segmentation. Overfitting, where the model performs well on training data but fails to generalize to unseen data, can result from such shortcomings.

To address these challenges and ensure robust segmentation results, it is imperative to curate a large and diverse training dataset. This dataset should encompass various conditions, including different image modalities, imaging protocols, and anatomical variations. Expert annotation of the training data is crucial to ensure accurate labels and minimize segmentation errors.

Furthermore, maintaining a balanced dataset with an adequate representation of each class is essential to avoid bias towards the dominant class. In medical imaging applications, imbalanced datasets can lead to inaccurate diagnostic or treatment decisions with potentially serious consequences.

Variations in lighting conditions necessitated the partitioning of images and separate processing of each section. This requirement implies the creation of multiple networks, each dedicated to a specific section. Additionally, differences in skin color posed a challenge, requiring a secondary separation, effectively doubling the number of required networks.

In summary, the quality, quantity, and diversity of training data are indispensable for the U-net network to learn the necessary features and patterns for accurate image segmentation. Overcoming challenges related to imbalanced datasets and variations in lighting conditions is essential for achieving optimal performance in vitiligo segmentation.

Training Process:

- The training process began with 60 patched images of size 128x128, yielding promising results despite the limited dataset.
- Additional data accelerates the network's training speed.
- Comparison: The network trained with 60 images reached a similar state after 500 epochs as the network trained with 60 BW images after 100 epochs.

The network trained were of the following categories:

- BW images: The network trained with BW images
- RGB images: The network trained with Color images

- RGB images with changed channel and added 15 pixels to the red channel
- RGB images and a sketch overlay.

The neural network was trained using these modified RGB images, allowing it to learn and classify the distinct skin regions accurately. By incorporating this preprocessing step, we aimed to enhance the network's ability to identify vitiligo-affected areas and improve overall diagnostic performance.

UNet with B/W images

In the next phase of our research, we explored the utilization of the UNet model with the data patches converted to black and white (B/W).

Nevertheless, the trained network exhibited notable success in detecting vitiligo regions for certain skin colors, returning reasonably accurate results. The findings from this phase of the research indicate the potential of leveraging the UNet model for vitiligo segmentation, particularly in cases where skin color is well-represented in the training data.

To further improve the performance of the network, we pursued two approaches. Firstly, we opted for a continuous improvement strategy by increasing the availability of ground truth data. This involved creating additional patches, thereby expanding the diversity and quantity of training samples. By continuously enriching the training dataset, we aimed to enhance the network's ability to generalize and accurately segment vitiligo regions.

Additionally, we explored the option of utilizing color images as training data. The previous phases of our research primarily focused on working with images in the HSV format, which provided a good match for the skin color and facilitated training on the corresponding HSV values. However, by incorporating color images into the training process, we aimed to incorporate additional visual information that could potentially enhance the network's performance, particularly in scenarios where the distinction between pigmented and non-pigmented skin is less prominent.

In the following sections, we will delve into the continuous improvement efforts and the integration of color images, evaluating their impact on the accuracy and robustness of the UNet model for vitiligo segmentation.

RGB Images and a Sketch Overlay

Overlaying an RGB color image with a sketch overlay can be a valuable approach to enhance the detection of edges, which are crucial for diagnosing vitiligo. The strong edges between non-pigmented and pigmented skin provide important differentiating features. By employing a sketch algorithm that generates a negative image and

subtracts the differences from the blurred image, we can effectively identify the edges within the image. Incorporating this information as additional training features for the network can improve its performance.

To generate sketch overlays for the RGB color images, a sketch program was developed using the OpenCV library in Python. The program consisted of the following steps:

1. Splitting the input image into its color channels (blue, green, and red).
2. Increasing the red channel intensity by adding a constant value of 15 to enhance the edges.*
3. Merging the modified color channels back into the image.
4. Converting the image to grayscale using the cv2.cvtColor function.
5. Generating the negative image using cv2.bitwise_not.
6. Applying Gaussian blur using cv2.GaussianBlur to smooth the negative image.
7. Inverting the blurred image back to obtain the original sketch image.
8. Combining the sketch image with the original image using cv2.divide and scaling the resulting image.
9. Merging the sketch image into a 3-channel image using cv2.merge.

The resulting sketch overlay provided enhanced edge detection for the diagnosis of vitiligo. The sketch program was applied to a dataset of RGB color images during the training process, which consisted of 600 images and ran for 12,000 epochs. The effectiveness of the sketch overlay technique was evaluated based on metrics such as accuracy, precision, and recall, yielding promising results. The prediction of a test image proved high accuracy. The usage of the patient data showed some diffraction from the actual correct vitiligo patterns. Compared to the RGB Network we found a small divagation in the results. The networks detect some areas different from each other. Special pigmented skin is causing issues when it comes to detection.

As a variant, we can replace the range for the red channel by adding a brightness value of 2 and a contrast value of 1. This will increase the image details and create sharp borders for the pigmented and non-pigmented areas, which is useful as feature enrichment to the original image information.

Using color images for training

The utilization of color images for training the network was explored to improve the accuracy of vitiligo segmentation.

For our research, we utilized a dataset consisting of patches of vitiligo extracted from patient images. Each patch had a size of 128x128 pixels and was accompanied by a corresponding mask. To streamline the training process and reduce computational complexity, the dataset was saved in grayscale format instead of the original three-channel color images. This approach offered enhanced accuracy and reduced training time compared to training with full-color images.

Head Image Processing for Vitiligo Research

The following steps need to be taken to extract the head of the image and to evaluate it against the AI Network.

Neuronal Network Architecture

The AI algorithm extracted the face from the clinical picture and then created a vitiligo mask. Next, it generated the convolutional neural network (CNN) training and testing data by using the TensorFlow Keras library. Patches of the images were created to generate training data for the CNN. After testing the performance of the network with different patch sizes, a 128x128 size was chosen. In total, 15,000 image patches were created from the clinical pictures used for the training set (80% of the 100 facial vitiligo images).

Data Generation, Evaluation, and Data Analysis

The image containing the patient's head was used for processing. As pictures taken with UV flashes produced some shadows on the pictures, the image was processed in two patches: one from the left and the other from the right. The mean of the patches was used to estimate the difference in lighting. There are only limited possibilities to repair images that are under or overexposed. Where overexposed images the information is gone the underexposed images can be enriched. For the clinical analysis of the image it might be sufficient but for AI image analysis the image quality should be improved.

Locating Background

Background location was necessary to separate pixels that were connected to the patient. HSV stands for Hue (color/hue), Saturation (saturation), and Value/ Luminance (lightness/luminosity, not to be confused with brightness). Hue sets the position of the color on the color wheel (from 0' to 360'), saturation is the percentage of saturation (0% to 100%), and value is the percentage of luminosity (0% to 100%). To detect all or most skin tones, a set of images was processed and the upper and lower boundaries were adjusted. To collect human skin with as few false positives as possible, we set the HSV model as described below. Owing to hair color and other confounding factors, an accuracy of 100 % cannot be achieved.

Defining HSV Thresholds
lower_threshold [0, 48, 80]
upper_threshold [20, 255, 255]

Background subtraction created a mask for the pixels that fell within the boundaries. By removing the other pixels and setting them to black, a color image of the remaining skin can be obtained. The background by setting a skin mask will work when backgrounds have a different color.

Locating Faces

The image produced in the previous step was analyzed with Mediapipe Python (https://google.github.io/mediapipe/getting_started/python.html). It allowed us to detect faces and establish facial landmarks. Alternatively, one could use dlib (https://pypi.org/project/dlib/), which is a library for image recognition but is less accurate than Mediapipe. Using landmark detection, i.e. the location of essential points of interest, we were able to locate typical features of the image such as the eyes and nose. Figures 1a and 1b show the original image and the location of the face, respectively.

The information was then used to modify the image if it was not centered. The position of the nose and the left and right sides of the skin borders were used to split the image along the nose line so that both parts were of equal size. This allowed the difference in exposure during processing to be taken into account. Figure 1c shows the result: a combination of the original facial image concatenated with the vitiligo mask, revealing the superposition of the edges of the vitiligo on the original image.

Head Size

We then used a mask containing information about the position of the face and placed an ellipse around the remaining pixels. The total head size to be used to assess the extent of the vitiligo could then be calculated.

Calculating the extent of vitiligo

In the event of lighting differences, the image was split and each side was processed with different parameters.

Defining HSV vitiligo range
lower_threshold.[40, s_min, 61]
upper_threshold. [178, s_max, 255]

The correct skin tone was defined before applying the AI. Owing to the UV lighting, the skin had a specific color and the non-pigmented areas were more exposed.

Based on the HSV color schema and especially the saturation channel, the lower S channel marked the start of the area of vitiligo and the upper S channel ended it. Therefore, we used an iterative approach by changing the lower S channel value (s_min) until the pixels in the image were no longer visible. This step provided information about skin tone. The algorithm created a mask that it then superimposed onto the face and vitiligo lesions. In this way, we were able to calculate the percentage of white pixels classified as vitiligo based on skin color. The mask generated was used to create a dataset for Machine Learning.

Machine Learning

A supervised learning algorithm was used, where the supervision was performed by automatic processing. This was achieved by taking 128 x 128 patches out of the image and the mask. Next, a sequential CNN and a UNet were developed. The CNN had an accuracy of 93 %. Instead of using the ReLu (Rectified Linear Unit) function, we used a sigmoid function, i.e. a specific form of a logistic function.

Using the UNet extended the training time to about 5 min per epoch, and an accuracy of 76 % was reached by using black and white patches. The detailed performance of the networks can be found in the supplementary materials. The approach was as follows: first, an image was separated into patches and each patch was then classified by the UNet as either pigmented skin or vitiligo. If the UNet prediction was over 70%, the patch was counted as vitiligo. Otherwise, the patch was considered normal pigmented skin. This reduced the number of vitiligo pixels in the result. Therefore, CNN provided help with the decision regarding whether the area was more likely to be normal pigmented skin or vitiligo. The model thus demonstrated its high level of confidence in differentiating normal and vitiligo skin.

The head of the patient was selected as the main focus of the analysis due to its prominence and high visibility. In public settings, the head and hands often attract a significant amount of attention. Additionally, isolating the head from the rest of the image provided several advantages, including noise reduction and mitigating the challenges posed by complex lighting conditions.

To determine the appropriate network for each skin type, a technique called histogram backtracking, introduced by Michael J. Swain and Dana H. Ballard in their paper "Indexing via color histograms," was utilized. Darker skin patches were assigned higher weights (e.g., multiplied by 7), while lighter skin patches were assigned lower weights (e.g., multiplied by 1). By considering the weighted sum of pixels returned by each skin patch, a more accurate indication of the correct skin type could be obtained[50].

Use 2 Phase processing

The concept of employing a 2-phase processing approach emerged from the idea of utilizing the automatically generated mask to validate and determine if an area is pigmented or non-pigmented.

First Phase: In the first phase, the objective is to calculate the distance between high and low values and determine whether the skin is pigmented or non-pigmented. The steepness in the second phase serves as an indication of pigmented skin. However, several challenges arise, such as issues with white underwear or incorrect identification when the background or foreground is erroneously classified as a high or low value.

To establish the S min value, we consider a range that is not indicative of normal skin, exceeding the average skin tone in the image. However, a boundary problem exists in this scenario as well. If the estimated base S min value is excessively high and no extreme light areas are found, it can be assumed that the skin is normal. The upper level for S max can be set to approximately 42 as a rough approximation. Some images may require special settings for H max and V max values. H max controls the pigmented skin color range and its value varies for each image and patient. V max represents the very bright pixels in the image and can be used to determine the skin type. For instance, a V max value in the range of 250 to 255 indicates a very white vitiligo region or one that is well-protected from UV light. This information serves as valuable decision-making criteria for further processing.

One challenge encountered with this approach is the presence of light reflections that can contradict the correct S min location. The algorithm performs accurately in images without underwear or lighting issues. However, in images with such issues, the accuracy is significantly reduced as the algorithm struggles to capture the vitiligo patches accurately. To mitigate these challenges, it would be necessary to incorporate H min and V min boundaries, which are challenging to calculate.

It showed that only by looking at the skin color some extent the vitiligo region can be localized but in an image with different lighting conditions on the body, this technique alone is not feasible for the vitiligo localization. Certain non-pigmented areas can not easily be selected by changing the S channel values. Those areas involve the combination of H and V values as well and it makes it very difficult to create the settings by thresholding without any additional information.

Additionally, determining the correct skin value becomes even more difficult when the entire patch consists of vitiligo-affected skin, lacking any reference or clear boundary with normal skin. This issue also applies to the breast area in upper body

images, where the absence of pigmentation is natural due to minimal sun exposure. In such cases, if we know the standard pigmented skin color value, we can make an educated guess based on whether the patch's value matches that of pigmented skin.

Areas in the shade present further challenges, as they alter the values for S min and S max, falling outside the normal ranges. Applying AI techniques in the second phase can help improve the results in these problematic areas.

The idea arose to split the image into sections vertically or horizontally. This would allow us to cope better with the lighting situation. The problem that is introduced with this technique is the loss of context or to know what is the normal skin color. The local maxima of light skin can maybe be the normal pigmented skin color. There it is then not possible to distinguish between non-pigmented and pigmented skin.

Second phase: This involves processing the image vertically and horizontally. The image is divided into patches of size 128x128. Processing images of 3000x3000 pixels already takes approximately one minute. Although using increased patches would increase network training time, it would dramatically increase processing time. Each patch is then processed by the trained AI model, and if the prediction confidence is above 70%, the patch is used; otherwise, the original patch is retained. This approach enables better results in areas with uncertain conditions, such as over or under-exposed parts of the image.

For training the network, image patches and corresponding mask patches are used. They are split into patches of size 128x128. The quality or accuracy of the training was approximately 97%. To improve the training performance, the number of patches was increased to 600. And also complete pigmented patches were incorporated into the training. Using a batch size of 100 and 10000 epochs resulted in a dice_coef accuracy of 99.9%.

The UNet model is known for requiring a limited amount of data for training. However, when training with color images and accounting for different skin temperatures, the required amount of training data significantly increases. It is observed that training yields better results with more epochs and batch data sizes of around 100. Smaller batch sizes and lower epochs provide insufficient or no training results. The epochs used were about 3500 to 4000 with multiple restarts of the training on the latest training weights. This reduces the accuracy at the beginning of the training but resets and makes the network more robust.

The image format used for training is TIFF, and the initial results from the network seemed promising. However, when processing new data, particularly facial images,

the results were considerably low. The network appeared to be incorrectly identifying normal skin as vitiligo, whereas the non-pigmented regions were identified correctly. This could be attributed to the diversity of skin colors used for training, where the network aligns to the most common skin tone, resulting in incorrect color mapping. After the development of all the relevant parts, we were able to prove the system in a clinical study.

Clinical Study

We were able to test the implemented AI algorithm described before, in a clinical study with several patients and several expert physicians to compare the results of the AI with human expertise. The objective of this study was to build and evaluate a neural network allowing the detection of vitiligo lesions on photographs with a special interest in lesions located on the face as described in the paper “Evaluation of facial vitiligo severity with a mixed clinical and artificial intelligence approach”[48]. The following sections refer to the publication.

Patients:

This retrospective, non-interventional, single-center study was performed in the Department of Dermatology of the University Hospital of Bordeaux and was approved by the local ethics committee. All patients received a patient information sheet and gave oral consent Patients who accepted to be published provided written consent. Clinical pictures of the face were taken with the FOTOFINDER® body scan system combined with two high-output UV flashes. The photographs were standardized in a neutral position.

One hundred facial vitiligo pictures were selected, of which 80% were used for training and the remaining 20% for validation of the trained model. As a final test data set, 69 additional new facial pictures were used. They were taken from 59 adult patients of whom 10 were reevaluated 6 months after an intervention therapy. Three expert physicians estimated the proportion of vitiligo lesions on the face at two-time points. Vitiligo was assessed as 0% (no vitiligo lesion) to 100% (complete depigmentation). Inter- and intra-rater reliability were evaluated by comparing the scores between experts at each evaluation and between two different time points (test-retest) respectively. A two-way random model was performed for inter-raters and a two-way mixed for intra-raters. Absolute agreement and single-measure intraclass correlation coefficients (ICC) were also estimated. Inter-rater reliability was evaluated between experts and AI. Logarithmic transformation was performed for non-normally distributed data. The following guidelines for the interpretation of the ICC were used: < 0.4 was considered poor, 0.4–0.59 fair, 0.6–0.74 good, and ≥0.75 excellent (Cicchetti 1994).

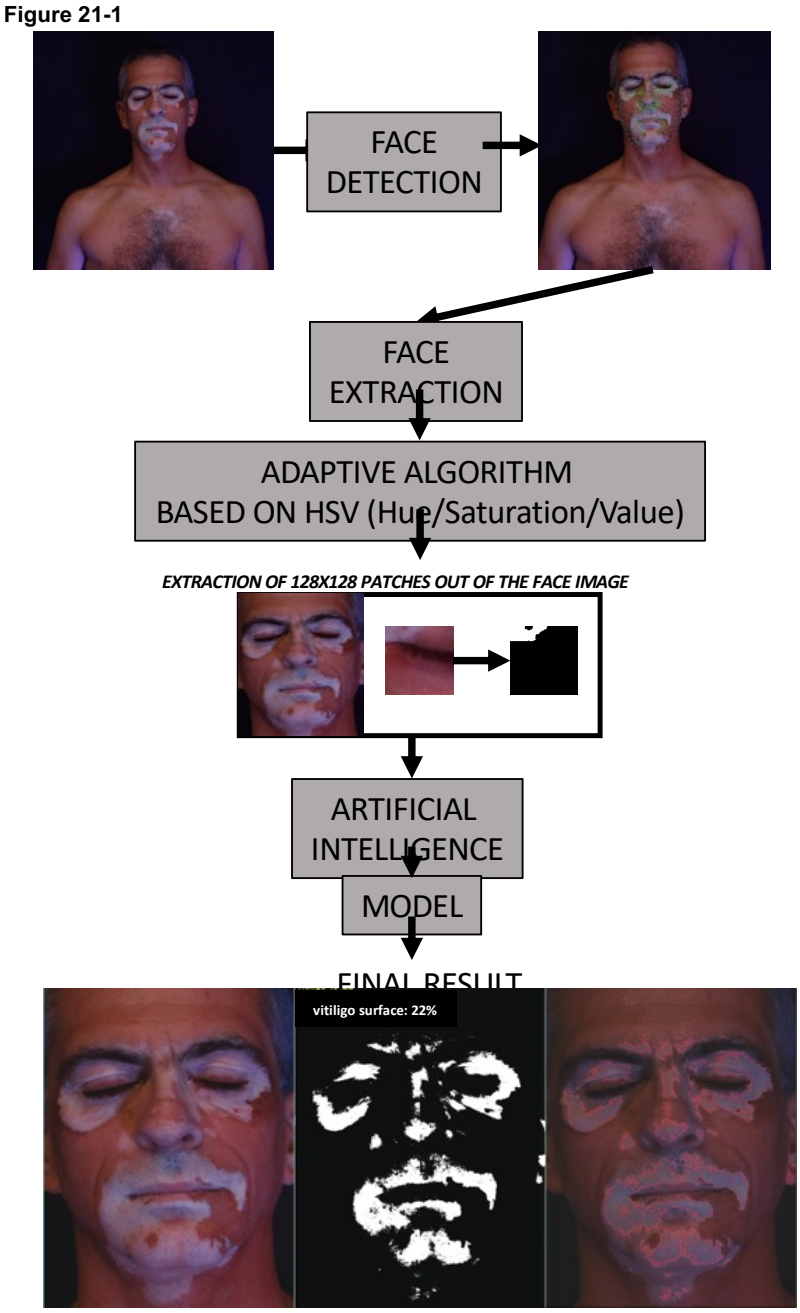
The training data for the head region was created using a mixture of pigmented, non-pigmented, and background/hair patches, as described earlier. This diverse training dataset yielded positive results, with the predictions of the trained AI network aligning closely with the estimations provided by physicians. To further enhance the accuracy of the network, the training data was repeatedly trained for 3000 epochs, resulting in improved performance.

The analysis revealed the need for different networks to accommodate different skin types. Vitiligo regions exhibit variations depending on the skin tone, necessitating specialized networks for images with lighter or darker skin. To address this, an inception module was employed between the initial HSV processing and the subsequent AI processing stage.

Assessing the severity of facial vitiligo with AI

The AI algorithm was designed as described in the above sections (Figure 21-1).

Figure 21-1) AI Algorithm



First, it extracted the face from a clinical picture and created a vitiligo mask (Figure 21a, b). Next, it generated the convolutional neural network (CNN) training and testing data. It was trained and tested using 100 vitiligo face images that were divided into a training set (80%) and a validation set (20%). After training, the CNN reached an accuracy of 93%.

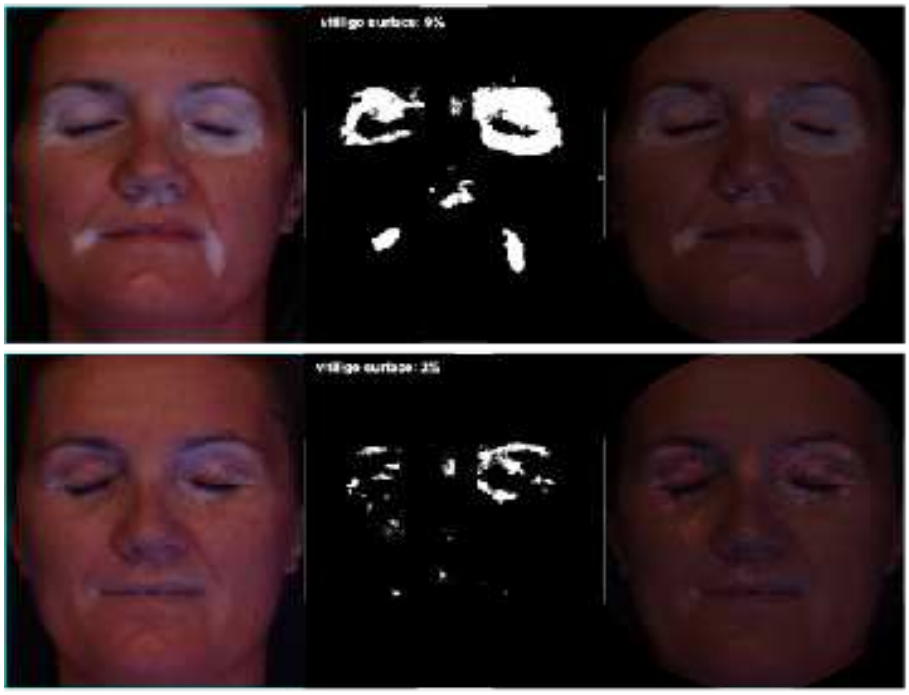
Figure 21) Image Processing

As shown in Figure 21c, the resulting picture was a composition of three images starting from left to right with the original image, the calculated vitiligo mask, and an overlay image of the original image plus the vitiligo mask image.

The vitiligo mask pictures contained a calculated percentage corresponding to the estimated depigmented area of the face ranging from 0% (no depigmentation) to 100% (complete depigmentation) (Figure 22a). The evolution of this percentage over time could be used in clinical practice to identify the positive or negative impact of treatments during patient follow-up (Figure 22b).

Figure 22) Result in Images for Vitiligo Processing

Figure 22) Result Images for Vitiligo Processing



a)Patient Analysis Results. b) Patient Analysis Results Time Series

Creation of Training Data for Body Images in Vitiligo Research
For body images, a distinct approach was required for mask generation and image processing due to the complexity introduced by different skin tones. Unlike head images, the processing of body images involves a sequence of steps:

Removal of Floor: The linoleum floor needed to be replaced or removed to avoid compromising the networks.

Removal of White Underwear: White underwear was removed to prevent interference with the networks.

Image Processing: All images were processed initially to facilitate the creation of a report with concatenated images at the end, requiring the implementation of a file management system.

Concatenation of Images: Images were concatenated at matching positions to provide a report with non-overlapping areas, necessitating combining the images in a specific order.

Handling Different Skin Tones: Due to varying skin tones in one image, different networks were utilized for different skin tones, requiring the creation and application of specific networks at the appropriate moment.

Patch-Based Training Approach
To ensure effective training of neural networks, a patch-based approach was adopted instead of using complete images. This approach offered advantages such as reducing the number of parameters and requiring less training data. However, the use of the previously discussed HSV value adaptation technique for creating patches was not feasible due to the difference in skin tones within one image.

Augmentation and Training Data Expansion
Three approaches were explored to enhance the patch data:
Augmenting patches with additional features, such as sketch information.
Adding to the red channel and changing the blue and red channels in the images.
Applying histogram equalization to ensure uniform color distribution in the images.
The networks were trained with patches modified with additional information, including Laplacian and Sobel edge detection overlays, increased red channel intensity, and histogram equalization.

Data Loading and Preprocessing
Image loading was performed using the OpenCV read () method or TensorFlow's image loading methods. Before data normalization, image data needed to be converted into a float32 array to ensure proper normalization. TensorFlow's methods facilitated immediate data normalization without additional conversions.

The UNet architecture demonstrated satisfactory results with a limited dataset, underscoring the importance of high-quality training data. Leveraging dynamic changes in color values enabled accurate estimations of vitiligo regions, serving as essential ground truth data during training.

Removal of Floor
The methodology for the removal of the floor in images containing feet and floor involves several steps to accurately isolate the skin area and remove the floor interference.

First, we acknowledge the challenge posed by the similarity in color between the floor and non-pigmented skin, which complicates the AI network's ability to distinguish between these regions.

To address this, we propose utilizing template-matching techniques to identify and eliminate the floor from the image. Template matching allows us to recognize patterns, such as the connecting floor and wall, which are indicative of the feet region.

Once the feet region is identified, we apply four removal templates to eliminate the floor, resulting in an image predominantly featuring the skin against a nearly black background. However, some regions of the skin on the feet may still be of similar color to the floor and may inadvertently get removed.

To address this issue, we use an eroded function to disconnect the remaining parts of the skin from the body. Subsequently, a contour algorithm is applied to remove the detached elements from the image. Finally, dilating the image helps in obtaining a more complete mask for the body while ensuring that the detached objects are effectively removed.

By following these steps, we aim to produce a refined image input for the AI network, facilitating more accurate analysis and segmentation of the skin area.

Removing white underwear
The methodology for removing white underwear from images poses several challenges due to the similarities in color between white underwear and certain areas of the skin, particularly those exposed to strong flashes. These challenges arise because AI networks may incorrectly interpret white areas as non-pigmented skin, leading to false positive results.

Inspired by techniques used for floor images, we developed a method to identify and remove white underwear from the image. By exploiting the distinct contour created by solid white areas, we implemented a filter to detect and isolate these regions based on a high value (V) in the HSV format image, typically around 253. This process enables us to create a mask highlighting the white areas in the image. Once the mask is generated, we apply an erode operation to remove smaller elements from the mask, such as white patterns or writing. This step helps refine the mask and distinguish between genuine skin areas and other white elements.

Subsequently, a dilate operation is employed to fill any gaps in the skin mask that may have been created during the erode operation. Additionally, this technique effectively removes small elements unrelated to the main object, such as patterns on the underwear or spots on the background.

By systematically applying these operations, we aim to accurately identify and remove white underwear from the image, thereby improving the accuracy of subsequent AI analysis and segmentation of the skin area.

UNet Training for Hair and Beard Removal
To mitigate the issue of white hair and beard sections, which often share color characteristics with non-pigmented skin regions in our patient images, we implemented a UNet-based approach. The UNet was trained to effectively distinguish and remove these unwanted hair areas from the images, thus enhancing the quality of our masks and aiding in more accurate vitiligo detection.

The UNet training process yielded promising results. We achieved an accuracy rate of 98% when training the UNet for 1700 epochs and subsequently improved it to 98% at 3400 epochs. This accuracy demonstrated the effectiveness of the network in distinguishing and removing hair regions.

Furthermore, to ensure robustness and consistency, we retrained the UNet four times, each time for 3400 epochs, and then applied the network to the patient images. This approach significantly improved the quality of the images, particularly in

terms of eliminating hair and beard areas that could potentially interfere with vitiligo detection.

The utilization of the UNet in this context not only exemplifies the potential of deep learning in image processing but also highlights its practicality in handling complex challenges like the removal of hair but needs improvement since the data is difficult to annotate its a challenging task.

Processing the images

The image processing methodology involves organizing and processing images obtained for a patient in a directory on the file system. To ensure proper handling and ordering of images, filenames within the folder must include the position of the image, either separated by a minus sign (e.g., XXXX-1.jpg) or enclosed in brackets (e.g., XXXX (1).jpg). Failure to adhere to this naming convention may result in incorrect image placement.

Front, back, and side images are provided and must be ordered sequentially, with unique numbering assigned to reduce complexity. The head image for the front is designated as 1, followed by additional images numbered sequentially up to 16, representing the last image taken for the patient.

Loose images not connected to a specific position are documented in a table along with the corresponding vitiligo percentage calculated. These images are assumed to be saved in the directory, with the directory name incorporated into the filename followed by a position number ranging from 1 to an unspecified upper limit. Other filenames not following this convention cannot be processed.

Each image undergoes processing, resulting in the generation of the original image, mask, and overlay of the mask and canny image, which are saved in an output directory. With all data saved, an image combination process is executed to assemble the images in the correct order, facilitating the creation of a PDF report.

Concatenation of Images

The methodology for image concatenation involves combining images from different parts of the body to produce a vitiligo count indicating the extent of vitiligo on the patient's body. This process requires careful alignment and concatenation of images without overlap. However, the positions where the images overlap are not fixed, adding to the complexity.

To address this challenge, sophisticated techniques are employed to accurately align and concatenate the images:

Use of Advanced Matching Algorithms: Utilizing feature-based matching algorithms such as Scale-Invariant Feature Transform (SIFT), Speeded-Up Robust Features (SURF), or ORB (Oriented FAST and Rotated BRIEF) provided by OpenCV. These methods are robust to variations in scale, orientation, and lighting. Those algorithms work more for general images with different objects and sharp edges. If the feature can not be located in the 2 images or be aligned the results are not satisfactory. In terms of concatenation of images with patient data containing mostly skin the features available for matching are very limited.

Implementation of RANSAC: Employing the Random Sample Consensus (RANSAC) algorithm to estimate a transformation matrix that best aligns the images. This helps handle outliers or inaccuracies in the matching process.
Manual Adjustment: In cases of complex structures and backgrounds, manual adjustment may be necessary to fine-tune the alignment. This involves selecting anchor points and adjusting positions, though automation is preferred. The same issue as in the above methods arises. We can not find enough key points to match the skin.

Utilization of Multiple Keypoints: Instead of relying on a single key point, use multiple key points across the overlap region for more robust alignment.
Preprocessing: Applying techniques like histogram equalization, contrast enhancement, or noise reduction to improve image quality before matching.
Data Augmentation: Creating augmented training data by applying various transformations (rotation, scaling, etc.) to images can help the matching algorithm handle variations in input data. Also, this algorithm using multiple key points did not work since the skin did not have many key points available also the correct alignment of one key point in one image to another key point in the other image did not match and was impossible to align since also the position of the object in the picture changed.

Template Matching: Template matching is a technique used to align and concatenate images, particularly when dealing with upper and lower body image concatenation. Template matching circumvents challenges associated with identifying key points or aligning borders precisely. Instead, it focuses on finding a matching region within the images, enhancing robustness. The template approach allows for the creation of templates based on either the skin mask or the original image, providing flexibility in matching both pigmented and non-pigmented regions if present.

Border Matching: Border Matching is an image alignment technique that establishes a connection between images by comparing their left and right borders. The algorithm involves identifying the maximum length line in the upper image and iteratively comparing it with corresponding lines in the lower image.

Assumptions are made to implement a robust algorithm, such as assuming a certain range of overlap in the images, especially for the head and upper body. The issue of the light floor affecting vitiligo detection and keypoint calculation for the feet is also considered.

One approach involves calculating the left and right borders of the images and finding matching borders in subsequent images. If the borders align, a potential match is identified. However, this method may not work for connecting the legs. The resulting combined images can be used for vitiligo calculations, with the front and back images each representing 40% of the body surface and the sides each representing 10%. The concatenated resulting mask can then be used to calculate the amount of vitiligo on the complete side of the body.

Template Matching

In situations where key point and border matching encountered challenges, an alternative strategy, known as template matching, emerged as a promising workaround. This approach offered a more reliable method for aligning images, particularly when dealing with the concatenation of upper and lower body images.

The Template Matching Process:

1. **Creating a Target Template:** To initiate the template matching process, a template was generated from the first 100 or 200 lines of the lower image. This template encapsulated a region of interest containing both skin and floor areas. This choice allowed for the exploration of more comprehensive mapping shapes.

2. **Template Placement in Upper Image:** The generated template was then placed within the upper image. Given the inherent overlap between the upper and lower images, it was reasonable to expect a corresponding region of overlap in the template's placement. This step sought to establish a connection between the two images based on a shared region.

3. **Identifying the Intersection Line:** Through template matching, the intersection point of the matching line between the upper and lower images

could be accurately located. This point signified the precise juncture at which the two images should be combined.

4. **Concatenating Images:** With the matching line identified, the upper and lower images were concatenated. This process ensured a seamless connection between the two images, yielding a coherent result.

The utilization of template matching introduced several advantages over conventional key point and border matching:

- **Robustness:** Template matching circumvented the challenges associated with identifying key points or aligning borders precisely. Instead, it focused on finding a matching region within the images.

- **Flexibility:** The template approach allowed for the creation of templates based on either the skin mask or the original image. This flexibility enabled the matching of both pigmented and non-pigmented regions if present in the template.

By adopting template matching, the image concatenation process gained a more robust foundation, ensuring accurate alignment and eliminating the limitations of previous methods.

Leveraging the Matching Line for Further Analysis
Once the matching line is established, the previously calculated images, including the original, the mask, the skin, and the canny image, can be segmented accordingly. The combination of these segmented images is reevaluated to calculate the proportion of non-pigmented skin versus the total skin content within the image.

The outcome of this process is presented in a structured PDF report, with each image occupying a dedicated page. The calculated Vitiligo percentage is provided beneath each image, offering a comprehensive assessment of the condition's extent.

For images that do not conform to the front, back or side positions, a table is included below the report, listing the image number, the corresponding image, and the calculated percentage.

If there is no matching line found in the upper image we need to add the second image below the top image. We have then to assume it is following and accept that we get duplicated skin areas.

Keypoint Matching

The possibility to match via key points is only possible if we have a large overlap in the images and also if there are some relevant key points to detect. Keypoint matching is an algorithm that works best with edges. The images provided do not contain many edges. That makes key point detection difficult. Also, the images contain non-pigmented and pigmented regions the mapping needs to be done on normal skin with other landmarks that are not detected because they are less prominent.

One example would be the leg region we would need to map the knees but due to the much more relevant floor data where more key points can be matched, the matching is not working. If the image contains underwear with writing the matching can be done more exactly. But that applies only to the middle part of the images provided.

What other option do we have if key points are not leading to a promising result? One alternative would be to look at the skin mask generated and to compare the borders.

Border Matching

An alternative approach to image mapping, known as Border Matching, offers a solution for aligning two images based on their left and right borders. By leveraging the information obtained from generated scan masks, this method aims to establish a coherent connection between the images' respective sides.

Algorithm for Border Matching:

1. **Maximum Length Line Identification:** The process begins with identifying the maximum length line within the upper image. This line serves as a reference for comparison.

2. **Iterative Comparison:** For each line index from 0 to the size of the image:
    - Find the maximum length line in the lower image.
    - Compare the two identified lines.

The implementation of Border Matching is influenced by variations in image angles. Due to the distinct contours resulting from different image angles, a degree of delta or fault tolerance is introduced into the distance calculation to accommodate these variations. Incorporating more certainty for Improved Border Matching to address the challenges posed by varying image angles, an AI-based approach can be introduced through template matching. The algorithm unfolds as follows:

1. **Template Creation:** Utilize the first 100 lines of the lower image to generate a template.

2. **Template Matching:** Match the generated template within the upper image.

3. **Matching Line Determination:** Identify the first matching line between the template and the upper image.

4. **Image Combination:** Concatenate the upper image and lower image two at the determined matching line.

The underlying assumption for Border Matching is the presence of a substantial overlap between images, spanning a range of 100 lines or more. As a result, alignment becomes feasible to a certain degree without necessitating morphological alterations to the images' orientation.

AI Model Training on the CURTA Cluster

The training of the dataset was performed on the SLURM Cluster, a high-performance computing system available at the Mathematics department of the University of Bordeaux. The cluster is equipped with GPU cards specifically designed for efficient image data processing. It includes nodes equipped with Intel Xeon CPUs and NVIDIA GPUs. Specifically, the cluster has nodes with Intel Xeon Skylake 6130 processors, each featuring 16 cores and a 22 MB cache, operating at 2.1 GHz. Additionally, the cluster is equipped with NVIDIA GTX 1080 Ti GPUs.

The Cluster is managed by a batch processing system. A job needs to be submitted into a queue and if the required capacity and CPU type. The job is processed and the results are created. The datasets provided. The creation of a pipeline approach to continuously retrain the Network was not possible to be implemented. We had to manually start the retraining and copy the trained Networks to the local machine.

Each image in the dataset had dimensions of 128x128 pixels with three color channels.

Computer time for this study was provided by the computing facilities of the MCIA (Mésocentre de Calcul Intensif Aquitaine)

During the training process, a batch size of 100 images was used. This configuration allowed for parallel processing and accelerated the training time. Each epoch, representing a complete iteration through the dataset, required approximately 4 seconds to compute.

To optimize the training performance, special settings were configured on the SLURM Batch system. These settings ensured efficient utilization of the available computing resources, including the GPU cards, to expedite the training process.

By utilizing the SLURM Cluster and its GPU capabilities, the training phase of the project was executed with enhanced computational efficiency, enabling the exploration of larger datasets and more complex models.

Specific GPU settings

The use of a GPU requires additional changes to be made. First, the TensorFlow needs to be installed with the TensorFlow-GPU version. This will install additional GPU libraries required for GPU support. In case there are more GPU cards installed the memory consumption needs to be set equally on both cards.

```
gpus = tf. config.list_physical_devices('GPU')
if gpus:
  try:
    # Currently, memory growth needs to be the same across GPUs
    for pain gpus:
      tf.config.experimental.set_memory_growth(gpu, True)
    logical_gpus = tf.config.list_logical_devices('GPU')
    print(len(gpus), "Physical GPUs,", len(logical_gpus), "Logical GPUs")
  except RuntimeError as e:
    # Memory growth must be set before GPUs have been initialized
    print(e)
```

The use of GPU is reducing the calculation time by about 80%. The required change for the training and validation data was the restriction of the GPU memory card. The batch size was a set to max of 20 with an image size of 256x256x3 compared to 100 images in 128x128x3. When the batch size is increased a different hardware is used to execute the software on the SLURM cluster. This leads to a change in the channel orientation. It is switching from channel first to channel last.

Continuous improvement

To further improve the performance of the network, we explored the use of the currently trained network and extended it with new training data. We leveraged the capabilities of the AI network to search for vitiligo and generated masks that defined the boundaries of the affected areas in the HSV color space. We utilized these masks to set the upper boundaries for the H, S, and V values. By using the AI-generated mask to determine the upper boundaries, we could examine the complete image area for any missing vitiligo patches. This approach was applicable when the mask was appropriately extended. However, if the mask contained a significant number of false positives, manual correction of the mask and subsequent processing of the corrected patches were necessary.

The algorithm implemented for this purpose is as follows:

1. Predict the presence of vitiligo on an image patch using the trained vitiligo mask.
2. Split the image into horizontal patches of size 128 x 128 in height and width.
3. Utilize the vitiligo mask to determine the lower and upper boundaries for the H, S, and V values.
4. Patch the newly generated mask into 128 x 128 patches and store them in a designated directory.
5. If a sufficient number of patches are available, initiate the model training process and add the patches to the training data.

By incorporating this algorithm into our workflow, we aimed to augment the training data and enhance the performance of the network. The additional training data derived from the extended masks provided an opportunity to further refine the model's ability to detect vitiligo accurately.

Results and Discussion

Challenges Encountered in Preclinical Embryo Models

By combining various imaging techniques with AI algorithms, the analysis of pigmentation in preclinical embryo models can provide valuable information for understanding biological processes and improving dermatological diagnosis and treatment.

The utilization of confocal microscopy, fluorescence microscopy, and brightfield microscopy, along with image analysis software and AI algorithms, enables a comprehensive exploration of pigmentation characteristics and their implications in the context of dermatology and related disciplines.

The obtained result images based on the overlay of contours created showed some interesting details. The structures revealed some additional information about the texture of the surface that was at first not visible. The amount of provided data did not allow to creation of a network nor conduct a vital set of training and validation data.

Challenges Encountered in Image Analysis and Methodology

In the Introduction to Retina Image Recognition, we encountered that extensive research has already been conducted in the field of retina image recognition [59]. However, research in this area was halted due to the wealth of existing studies and resources. There were several implementations with trained networks already existing and closed before going live. The area of diabetes detection with retina analysis was completely covered by those research groups and left little to no enhancements.

Challenges Encountered in In Vivo Mouse Models and Digital Pathology

In the Introduction to In Vivo Mouse Models, we aimed to separate the blood vessels of the image.

The created tool adjusts the image to allow the selection of the blood vessels easily. We encountered difficulties in finding the right combination or color range. The images showed quite a similarity in colors the exact creation of a mask was not possible. We used the mask but had to manually adjust the training data set see Figure 2-1) for details.

Vivo Mouse images:

- After 100 epochs:

  7/7 [==========] - 3s 427ms/step - loss: 0.7167 - dice_coef: 0.2855

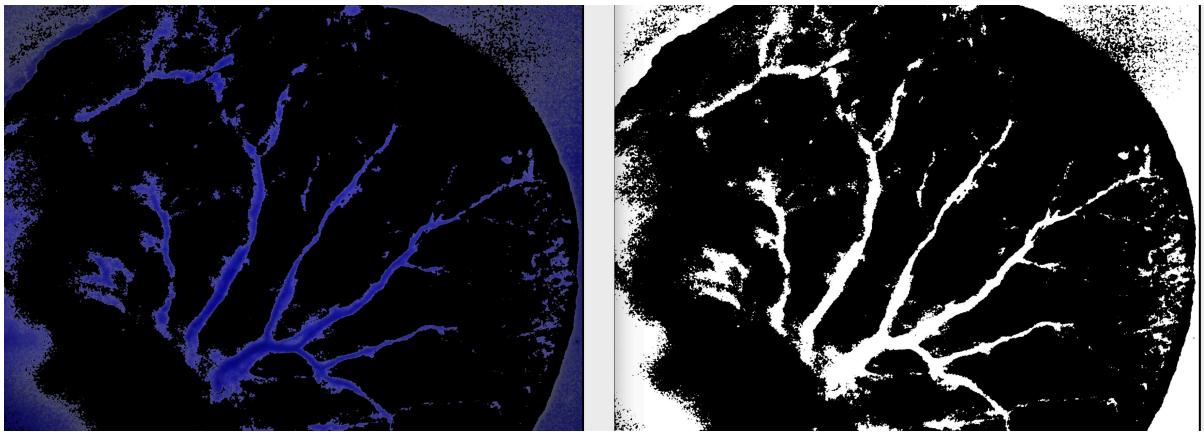
  - Accuracy: 28%
- After 600 epochs:

  7/7 [==========] - 3s 427ms/step - loss: 0.4399 - dice_coef: 0.4600

  - Accuracy: 46%

The UNet training process was initiated using the limited data available. However, the training results did not progress as expected due to the insufficient quantity of data. This limitation hindered the network's ability to generalize effectively, resulting

Figure 2-1) Mask generation vivo mouse images



in suboptimal performance. Addressing this challenge is paramount for future endeavors, as increasing the diversity and volume of training data could potentially yield more robust and accurate results. Moreover, exploring techniques for data augmentation or incorporating transfer learning strategies may offer avenues for mitigating the impact of limited data on network performance.

Blood Vessel Analysis for Cancer Prediction

One noteworthy endeavor involves automating the analysis of mouse ears for cancer prediction, a task traditionally performed manually. This process initiates with data analysis and segmentation, distinguishing the foreground (blood vessels) from the background. Inspired by human context and memory, computer graphics methods come into play, enabling the exploration and analysis of images [35].

The localization of blood vessels is achieved by applying threshold techniques and computing background and foreground information. The use of traced vector graphics libraries, such as Pypotrace, allows for the overlay of these graphics onto in vivo mouse model images, greatly facilitating the visualization and analysis of blood vessels [36][37].

Acknowledging Limitations

It is important to note that despite these innovative approaches, our research in the realm of in vivo mouse models and digital pathology encountered significant challenges. The availability of data proved to be insufficient, and we observed

substantial progress made by other dedicated research groups in these fields. As a result, we found it prudent to halt our endeavors in this particular direction.

Vitiligo Research Outcome

In the Introduction to Vitiligo Research, the patient images provided showed not from the beginning all obstacles. They revealed their existence during the process of the analysis. Due to the more prominent position, the focus was moved to the head. The process then involved the location of the head and to extract the head from the image. Here the first challenge was the location of the head. To separate the background. The creation of masks for AI training and to cope with difficult light situations. With a broader view of the complete body, we encountered more challenging aspects as there are different skin tones in one image the use of white underwear as well as the floor being of the same color as non-pigmented skin. But for all image processing the computer graphics are the base and need to be looked at at first. This contains the image loading and processing of images as well as their manipulations.

Computer Graphics in Vitiligo Detection

Several Computer Graphic techniques were used for vitiligo detection. Beginning with skin detection and object detection to find the object of interest. We discovered the head and hands with some limitations. The color scheme to use was vital for the remaining process.

Color Space Analysis

Various color schemes exist in computer vision, such as RGB, YCrCb, and HSV. RGB stores colors in three channels: Red, Green, and Blue. While the blue channel is a strong candidate for vitiligo skin recognition, using this channel alone is often insufficient to locate vitiligo regions accurately. We employed the HSV (Hue, Saturation, and Value) color space to generate skin masks for the images in our dataset. This approach enabled us to effectively isolate skin regions based on saturation and hue values. The skin masks obtained from the HSV-based skin detection method facilitated the calculation of the total amount of skin present in each image.

Upon analysis, we observed variations in skin tone across the dataset. Skin tones were characterized by distinct ranges of hues and saturation values, allowing for accurate segmentation of skin regions. The flexibility offered by the HSV color space allowed us to adjust the hue, saturation, and value components to fine-tune our skin tone detection algorithm. This adaptability proved beneficial in accommodating different lighting conditions and skin tone variations present in the images.

Furthermore, statistical analysis conducted on the skin tones revealed meaningful insights. By calculating statistics such as mean and standard deviation for each component of the HSV color space, we gained a deeper understanding of the distribution of skin tones within the dataset. This facilitated the detection of trends and anomalies in skin tone distribution, contributing to the overall analysis.

In addition to detecting skin regions, our approach allowed for the identification and assessment of vitiligo regions within the images. We observed varying degrees of vitiligo present across the dataset, with the extent of vitiligo lesions ranging from minimal to extensive. The analysis provided valuable insights into the distribution and appearance of vitiligo, aiding in the characterization of the condition within the dataset.

The adoption of HSV-based skin detection and subsequent vitiligo region identification offers a robust and reliable approach for analyzing and quantifying vitiligo-related skin anomalies. This methodology enables accurate characterization and monitoring of vitiligo progression, facilitating comprehensive research and potential advancements in treatment and intervention strategies.

For the provided images, acceptable results were obtained by defining the skin color range as follows:

```
hsv_img   =  cv2.cvtColor(cc_img, cv2.COLOR_BGR2YCR_CB)
skin_min  = np.array([70  , 113,   90], np.uint8)
skin_max = np.array([255, 180, 190], np.uint8)
skinMask = cv2.inRange(hsv_img, skin_min, skin_max)

viti_min = np.array([190,  130, 120], np.uint8)
viti_max = np.array([250, 190, 190], np.uint8)
vitiMask = cv2.inRange(hsv_img, viti_min, viti_max)
```

Here is an example of skin detection using the RGB format:

```
skin mask = np_multi_and(frame_r > 205, frame_g > 151, frame_b >
241,np_multi_max(frame_r, frame_g, frame_b) - np_multi_min(frame_r,
frame_g, frame_b) < 60, abs(frame_r - frame_g) < 55, frame_r >
frame_g, frame_r > frame_b)
```

Similarly, skin detection can be performed using the YCrCb format:

```
YCRimage = cv2.cvtColor(cc_img, cv2.COLOR_BGR2YCR_CB)
arrMinYCrCb  = np.array([  26, 103, 110], np.uint8)
arrMaxYCrCb = np.array([255, 176, 175], np.uint8)
```

skinMask = cv2.inRange(YCRimage, arrMinYCrCb, arrMaxYCrCb)

By providing code snippets in both RGB and YCrCb formats, we demonstrate the practical implementation of skin detection techniques. A combination of all three results will provide the best fit to obtain all skin colors in an image.

Using the skin mask obtained from HSV-based skin detection, we can calculate the total amount of skin present in the image. This quantification serves as the baseline for further analysis, particularly in the context of vitiligo calculation. By utilizing the information from all three channels (Hue, Saturation, and Value), we can identify vitiligo regions and assess their extent within the image.

In conclusion, leveraging the power of HSV color space and utilizing all three channels, we can effectively detect and analyze skin conditions such as vitiligo, contributing to a deeper understanding of the disease and its impact on patients.

Object Detection Results

In our investigation of various object detection methods, we encountered limitations when applying existing implementations in the clinical environment. The nature of medical images, characterized by non-solid backgrounds and variable lighting conditions, posed challenges for conventional object detection techniques.
Challenges with Background Noise

Our adaptation of object detection algorithms revealed significant challenges, particularly in scenarios involving high-resolution patient images. One prominent issue was the presence of patterned background fabric, which introduced considerable noise into the images. Despite attempts to blur the background, the woven fabric pattern persisted, leading to inaccuracies in separating the patient from the background. This noise not only interfered with the detection of the intended objects but also affected the quality of the detected boundaries, resulting in imprecise object delineation.

The application of blur to mitigate background noise presented additional challenges. While blurring effectively reduced background interference, it also affected the clarity of the object edges. As a result, the boundaries of the detected objects became less distinct, compromising the overall accuracy of the detection process.

Our findings suggest that the adapted object detection approach is most effective in scenarios with solid backgrounds that contrast distinctly with the objects of interest. In such cases, the absence of background structure facilitates accurate object detection. However, in clinical settings where background variability and noise are prevalent, further refinement of the approach is necessary to improve its applicability.

The challenges encountered in adapting object detection algorithms underscore the importance of considering the specific characteristics of medical images and clinical environments. As discussed in the color space analysis section, our ability to apply consistent skin tone settings across the entire image proved advantageous. Given that all skin tone colors within the image are presumed to belong to the same patient, this approach becomes more viable for comprehensive image analysis. By making slight variations to accommodate a wider range of skin tones, we can enhance the effectiveness of object detection algorithms across the entire image.

Face Detection

Face detection plays a pivotal role in vitiligo research, as the face represents a highly visible area of the body that is often exposed to sunlight. Despite the significance of facial detection in vitiligo studies, existing AI networks encounter challenges in accurately identifying vitiligo-affected areas on the face.

The majority of AI networks developed for vitiligo detection have been trained on ground truth data consisting primarily of images of pigmented faces. As a result, these networks exhibit high accuracy in detecting pigmented skin areas but struggle with depigmented areas characteristic of vitiligo. Discrepancies arise due to differences in key points and shading variations specific to the face.

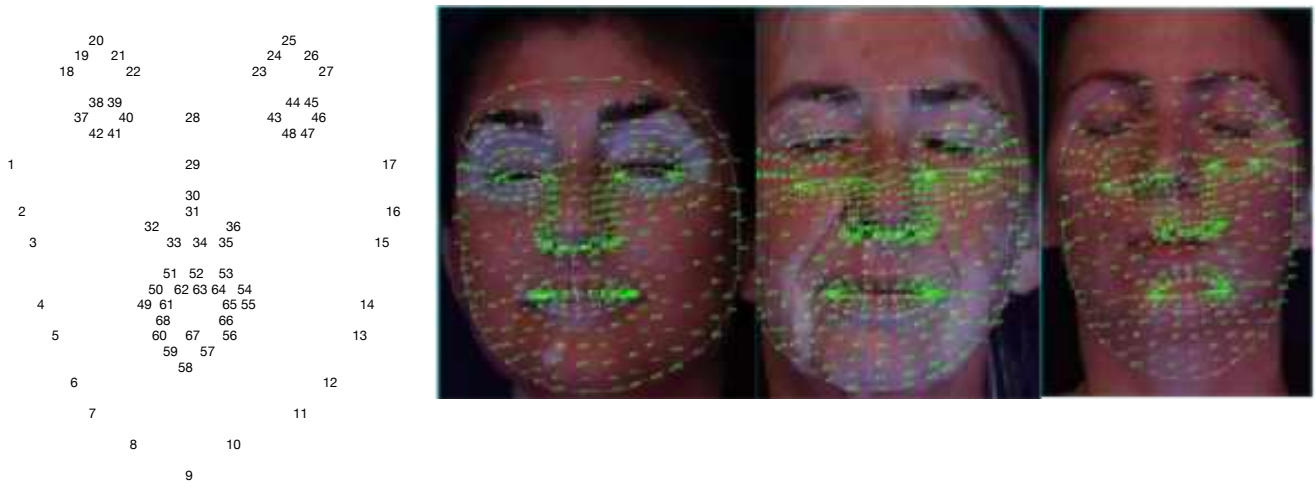
Variations in lighting conditions, skin texture, and facial features further compound the challenges faced by AI networks in accurately identifying vitiligo-affected areas on the face. These factors contribute to inconsistent detection results, ranging from minor discrepancies to substantial inaccuracies.

To address these challenges, one approach involves selecting multiple regions within an image for detection. However, this method may encounter limitations, particularly when attempting to detect facial features in non-traditional locations, such as the breast area. Despite these potential limitations, the network might still attempt to identify facial features based on contour patterns resembling a face.

The detection of facial landmarks is a crucial aspect of face detection. These landmarks, which include specific points such as the eyes, nose, and mouth, help locate and identify more specific body parts. However, incorrect landmark detection can occur, especially when normal facial structures are absent. This inaccuracy can be attributed to various factors, including lighting conditions or skin conditions like vitiligo and other autoimmune diseases.

Figure 23A) illustrates the facial points matched to key points on the image, showcasing the mismatch on vitiligo-affected skin. Three patients' faces were utilized to demonstrate the challenges encountered in accurately detecting vitiligo on the face.

Figure 23) Facial Points and Face Detection



A) Facial Points B) Facial Points on Vitiligo Faces

A) Facial Points B) False Face Detection Samples

As depicted in the first image of Figure 23B), the face is recognized correctly, and the eyes, nose, and mouth are appropriately located. However, in the middle image, the model struggles to accurately detect the eyes, while the nose and mouth are still detected correctly. The rightmost image illustrates the worst-case scenario, where the eyes, nose, and mouth are all inaccurately located. This failure can be attributed to the fact that the vitiligo patterns present a distinctive pattern that is perceived by the model as more closely resembling a face than the actual facial features. Since the model is primarily trained on images of individuals with normal skin, it struggles to adapt to the unique patterns exhibited by vitiligo patients.

Consequently, the accuracy of face detection in vitiligo patients is compromised due to the disparity between the training data and the specific characteristics of vitiligo-affected faces. This limitation highlights the need for developing specialized models or modifying existing ones to account for the unique patterns and features associated with vitiligo. By addressing these challenges, we can enhance the accuracy of face detection in vitiligo patients, enabling more effective analysis and advancing research efforts in this field.

Face Overlay

The methods and techniques discussed provide valuable insights into the distribution and proportion of non-pigmented regions, particularly in vitiligo-affected areas. However, the process of overlaying facial features poses significant challenges due to several factors.

Unlike objects with static positions, such as book covers, facial features are dynamic and subject to change over time. Variations in hair, facial expressions, and vitiligo patterns contribute to the difficulty of finding consistent key points for matching. Consequently, the accuracy of facial overlay is compromised, leading to unreliable results.

The inability to achieve accurate facial matching highlights the challenges of applying this approach in a clinical environment. Incorrect matches can lead to erroneous conclusions and undermine the reliability of the method for clinical use. As such, it becomes evident that the current approach lacks the tolerance and robustness required for practical application in clinical settings.

Given the limitations encountered with facial overlay, alternative strategies may need to be explored. One such approach involves detecting and analyzing other prominent body parts, such as the hands. Hand detection has shown relatively good performance in identifying vitiligo patterns, albeit with its own set of challenges. By simplifying the detection process to focus solely on location and bounding box generation, more reliable results can be achieved, offering potential avenues for further investigation.

The challenges encountered in overlaying facial features underscore the complexity of analyzing dynamic and variable structures such as the human face. While current methods may fall short in clinical applications, ongoing research, and refinement of techniques may yield improved approaches for assessing vitiligo-affected regions in the future.

Morphology Results

The implementation of morphological analysis for vitiligo assessment yielded unsatisfactory results, primarily due to challenges related to the timing of image acquisition and the absence of matching points between images.

One of the key obstacles encountered in applying morphological analysis was the temporal difference between image acquisitions. As images were captured at varying intervals, ranging from weeks to months apart, the lack of temporal consistency prevented the establishment of corresponding points for matching. Consequently, attempts to align and compare images using morphological techniques proved futile.

Moreover, the ineffectiveness of morphological analysis hindered our ability to evaluate the efficacy of therapeutic interventions for vitiligo. Without reliable methods for comparing pre- and post-treatment images, it was challenging to discern whether the observed changes in vitiligo severity were attributable to the treatment or other factors.

The analysis of available data revealed that morphological analysis did not yield promising results in our context. Despite efforts to explore this approach, the absence of identifiable matching points and the inability to establish temporal coherence rendered the methodology ineffective for our purposes.

The limitations encountered in morphological analysis underscore the need for alternative methodologies in vitiligo assessment.

Dynamic Mask Generation Results

In our approach to vitiligo detection, a critical consideration was the determination of the upper S value (Smax) for adaptive HSV adjustments. By dynamically setting Smax based on the S lower value, we aimed to create a flexible range within which color variations associated with vitiligo could be accommodated. This method contributes significantly to the accuracy and adaptability of our vitiligo detection methodology.

Our empirical analysis involved closely examining the HSV color scheme and analyzing sample skin patches extracted from the provided images. Within these images, we identified seven distinct major skin colors, spanning from dark-pigmented skin to light non-pigmented skin. By scrutinizing the H and S values within these patches, we derived valuable insights into the appropriate boundaries for adaptive HSV adjustments.

The results of our analysis have implications for the development of ground truth data for AI-based vitiligo detection algorithms. Despite the marginally better accuracy of AI-based methods compared to the adaptive threshold generation approach, our adaptive method demonstrates promising results, especially considering its potential to serve as a reliable pre-selection tool for generating ground truth data.
Connecting back to the research question, our findings underscore the feasibility of developing an algorithm or AI for the automatic detection of vitiligo in patient images. While the AI demonstrated slightly superior performance compared to the adaptive threshold generation method, both approaches achieved good ratings. This was corroborated by a study involving comparison with evaluations by physicians and analysis of AI-generated results.

Although visual aids were not available, the charts provided in the study offer valuable insights into the performance of different raters and algorithms. These charts illustrate the agreement between raters and the AI, highlighting the effectiveness of the adaptive threshold generation approach in vitiligo detection.
In summary, our dynamic mask generation approach represents a significant advancement in vitiligo detection methodology, offering flexibility, adaptability, and promising results that contribute to the development of accurate and efficient AI-based diagnostic tools.

Vitiligo Calculations Results

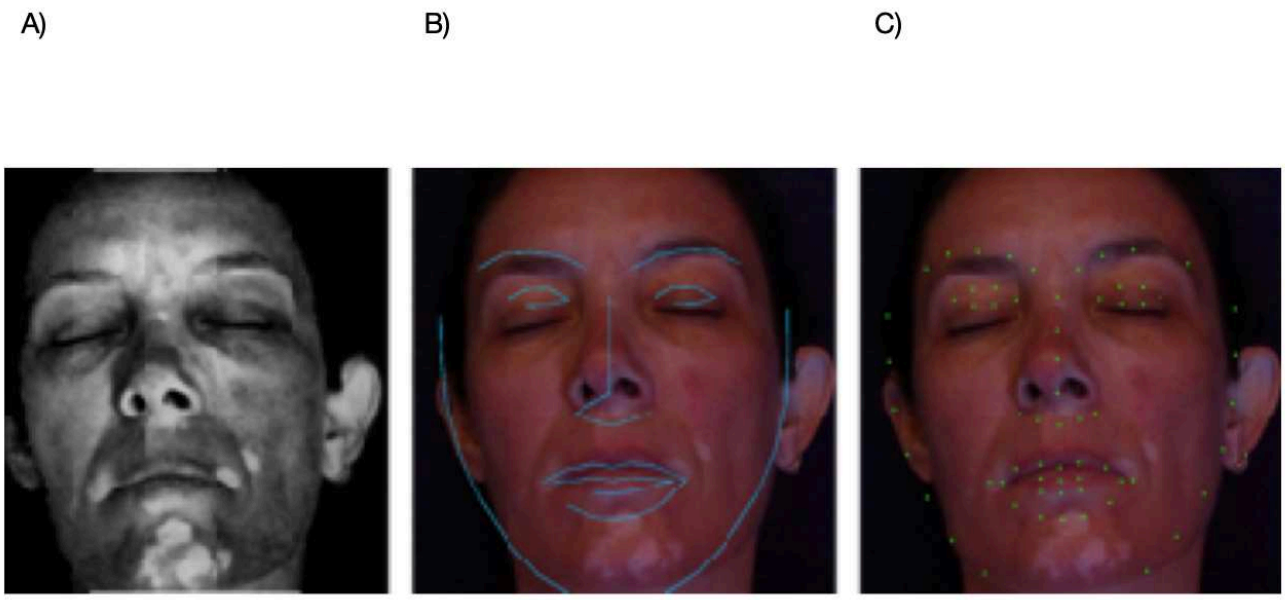
In our quest to quantify the vitiligo extent within patient images, we encountered challenges due to the absence of a reliable measurement system embedded within the images themselves. This led us to adopt a practical solution to estimate vitiligo amounts.

One approach involved leveraging the assumption that the size of the eyelid remains relatively constant across individuals. By measuring the distance between the eyelids in the image, we aimed to establish an applicable measurement scale. However, accurate positioning of facial landmark points required for this method proved challenging due to variations in face poses, expressions, and image quality.

Moreover, in cases where facial landmarks were not correctly identified or discovered at incorrect positions, the resulting measurements were inaccurate. Another proposed option involved using a marker of known dimensions positioned in the background to establish a reference for measurement. However, this approach was not feasible in the clinical setting. It might be a future requirement for place makers to allow cm-based metering.

As illustrated in Figure 24), attempts were made to locate the face and identify relevant facial points Figure 24A) for eyelid measurement Figure 24B). However, practical implementation revealed that this method was unreliable, with inconsistent detection of eyelids Figure 24C) and variations in image sizes leading to inaccurate pixel distance measurements.

Figure 24) Face Point Measurement



A) Original Image in BW B) Image Face Location C) Face Points discovered

In light of these challenges, we pursued an alternative method based on pixel counting. The total visible skin area in the image was considered 100%, and the percentage of non-pigmented pixels detected was measured relative to this total. While this approach may not provide precise square centimeter measurements, it offers a practical and reliable means of quantifying vitiligo extent.

The decision to opt for pixel counting was influenced by the feasibility and reliability issues encountered with other methods, particularly the inconsistent detection of facial landmarks by AI networks. By counting pixels, we were able to overcome these challenges and achieve consistent and meaningful results in quantifying vitiligo extent within patient images.

Image Processing and Thresholding

In our pursuit of improving vitiligo detection accuracy, we explored various image processing and thresholding techniques. Despite the potential of these methods, their integration into the vitiligo detection process posed significant challenges and yielded mixed results.

Determining Appropriate Thresholds: Locating suitable thresholds for image segmentation, especially under varying lighting conditions, proved challenging. The diversity of skin types and shading, coupled with fluctuating illumination, rendered threshold determination impractical for our purposes.

Multi-Otsu Threshold: While the Multi-Otsu threshold algorithm showed promise for segmenting pixels into different intensity classes, its direct application to entire images under varying lighting conditions resulted in imprecise outcomes. Despite attempts to improve accuracy by applying the algorithm to smaller image patches, discrepancies persisted. Although the Multi-Otsu Threshold provided indications for upper and lower thresholds, aligning them with the required Smin and Smax values proved challenging.

Edge Detection Algorithms: We experimented with edge detection algorithms, such as Sobel Feldman and Laplacian, to identify sharp edges characteristic of vitiligo regions. However, these algorithms struggled to differentiate between skin variations and background noise due to the complex nature of human skin. Skin tones exhibit subtle variations and lack clear, defined edges, making traditional edge detection less effective for skin detection.

Fast Fourier Transformation for Motion Detection: Despite its effectiveness in detecting motion and blur, Fast Fourier Transformation did not yield satisfactory results for vitiligo region identification. Skin tone variations, lighting differences, and inherent skin shades introduced unintended contours, making it unsuitable for our purposes.

Adapting to Different Skin Types: Handling variations in vitiligo regions across individuals with diverse skin tones required specialized algorithms. By implementing variable Smin values to calculate Smax, we achieved improved segmentation tailored to different skin types.

Histogram Backtracking for Skin Tone Identification: We employed histogram backtracking to identify predominant skin tones and locate vitiligo regions. However, determining the correct skin type posed challenges due to the presence of non-

pigmented skin and lighting variations. We adapted the solution to account for these challenges, although accurately defining skin color remained difficult.

Template Matching for Object Detection: Template matching initially showed promise for head and hand detection. However, challenges arose due to variations in human appearance, including different positions, orientations, and scales. Creating templates for each variation proved impractical, and template matching alone was insufficient for accurately detecting humans in diverse poses.

Incorporating these techniques into the vitiligo detection process provided valuable insights into their potential and limitations. While each method contributed to our understanding, they also highlighted the complexities encountered in real-world scenarios, particularly about diverse skin colors, lighting changes, and variations in human appearance.

Gabor Filters and Texture Analysis

In our endeavor to detect vitiligo patches, we explored several edge detection techniques, including Sobel, Roberts Cross, Gabor, and Laplace filters. Figure 25 depicts examples of Sobel and Gabor filters. Sobel, Roberts Cross, and Gabor as well as Laplace. A Sobel filter with X, Y, and X+ Y is shown in Figure 25A). Samples of Gabor filters can be found in Figure 25B)

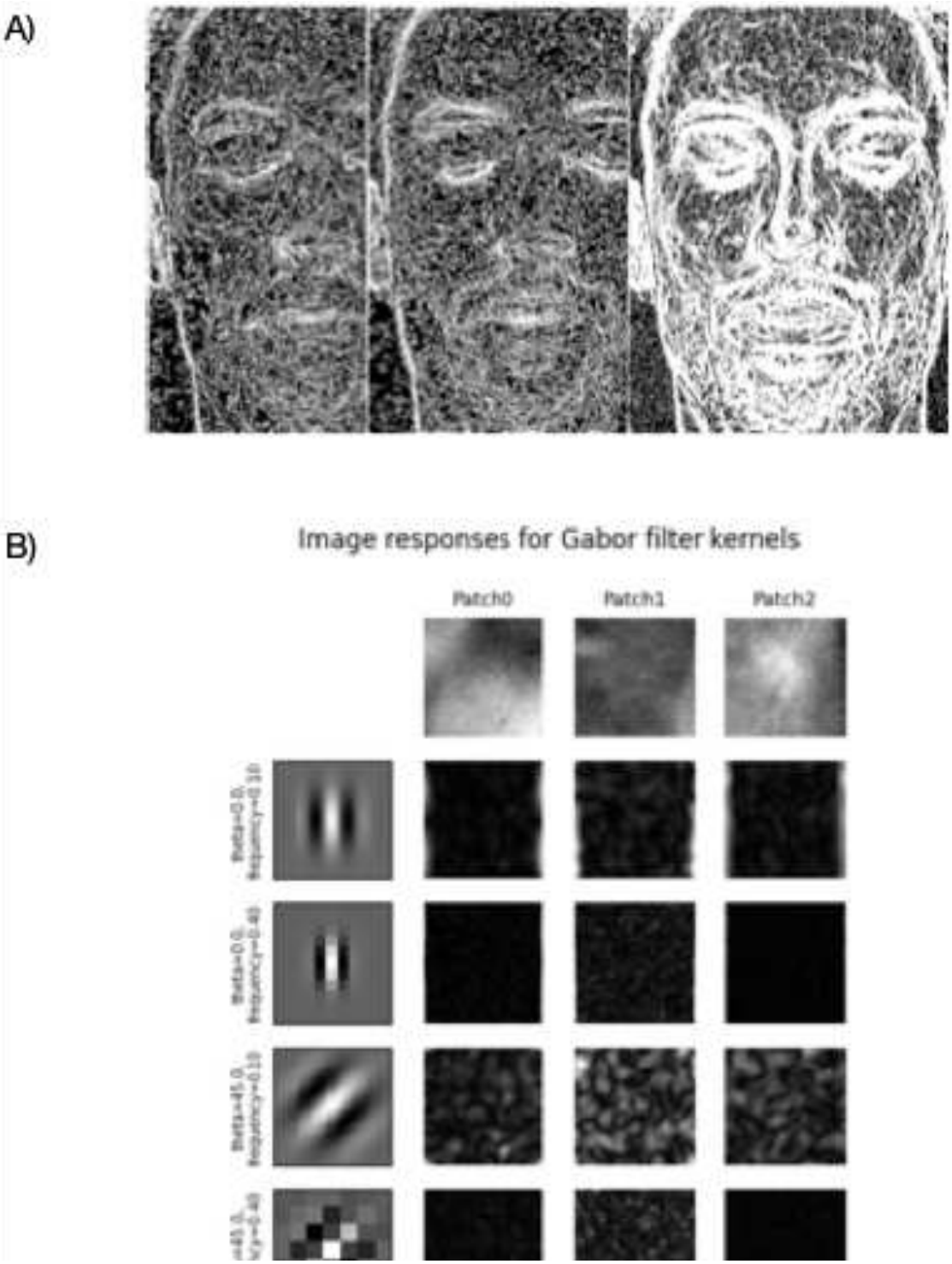
The use of Gabor filters yielded limited results in vitiligo detection. While effective for detecting sharp edges in materials like pebbles or bricks, Gabor filters struggled to distinguish subtle variations in skin pigmentation, often resulting in noisy output. The minimal difference in pixel values between pigmented and non-pigmented areas proved insufficient for accurate edge detection in this context.

Conversely, the Sobel filter demonstrated more promising results in highlighting the edges of vitiligo patches, facilitating accurate patch detection and localization. However, while effective, the Sobel filter can also generate patterns that provide limited additional information, contributing to noise in the output.

The Roberts Cross filter, although fast and simple, exhibited limitations compared to more sophisticated algorithms like the Sobel filter. It may miss some edges and produce noisier results, particularly in complex textures like human skin. Additionally, preprocessing techniques such as Contrast Limited Adaptive Histogram Equalization (CLAHE) combined with the Sobel filter generated excessive details, leading to reduced learning rates and accuracy of the UNet network. This combination failed to achieve precise and reliable vitiligo detection due to the overwhelming noise introduced by the filters.

Overall, while the Sobel filter proved effective in highlighting edges, the complexity and variability of human skin tones posed challenges for accurate vitiligo detection. The Sobel filter's ability to compute gradient magnitudes and emphasize edges in different orientations makes it a valuable tool in various image processing applications.

Figure 25) Sobel and Gabor Filters



A) Sobel Filter X,Y,and Combined. B)Gabor Filter on 3 Patches

However, in the context of vitiligo detection, the inherent variability of skin tones resulted in significant noise in the filters, rendering them ineffective for analysis. Despite its utility in other applications, the Sobel filter's performance was limited by the unique characteristics of human skin.

The outcome of Mask Creation

In our quest to automate mask creation for vitiligo detection, we explored various algorithms, each with its own set of challenges and outcomes.

Adaptive Mask Generation: Implementing an adaptive algorithm enabled us to create masks capable of accommodating different skin colors. Despite achieving accurate lower and upper limits for color channels, challenges arose in identifying healthy skin areas when patients exhibited approximately 80% vitiligo coverage. Furthermore, shaded areas around the nose posed difficulties, and potential solutions involving sliding window techniques were explored.

ANT Network Implementation: An initial attempt at using the ANT Network for mask creation proved inefficient and time-consuming. Despite efforts to refine processing boundaries, the complexity of the Python framework hindered parallel computation, resulting in unsatisfactory outcomes.

Hill Approach for Vitiligo: Exploration of the Hill Approach, which relied on the blue channel for calculations, encountered issues with reflective skin areas exhibiting similar values to vitiligo. Relying solely on the blue channel proved inadequate for distinguishing vitiligo patches.

Flood Fill Algorithm from Hill Approach: Building upon the Hill Approach, an algorithm incorporating flood-fill techniques encountered challenges in determining the correct delta value for intensity change, leading to inconsistent results across differently lit areas.

Moving Thresholds in Vitiligo Research: A dynamic approach known as "Moving Thresholds" showed promise but proved impractical for clinical use due to its time-consuming nature, particularly for patients with a large number of images. Additionally, issues related to overexposure persisted despite attempts to control thresholds.

Watershed Algorithm: Exploration of the Watershed Algorithm for separating vitiligo-affected and normal skin patches based on topographic features faced challenges in determining precise boundaries and exhibited lengthy runtimes.

Contours and Edges in Vitiligo: Direct application of contours and edges onto images added noise and failed to provide significant information without a final mask, limiting their utility in vitiligo detection.

Image Overlay and Time Series: Exploration of image overlay and time series analysis revealed potential for evaluating the progression of vitiligo treatment over time. However, challenges in matching landmark points and overlaying images taken at different times and angles hampered its practical application. Despite exhaustive attempts with various algorithms, including the use of the blue channel and landmark detection, we encountered significant challenges in automating mask creation for vitiligo detection.

Figure 26) Overlay of Images The process of overlaying images encountered challenges, particularly in matching landmark points. While it is possible to overlay two facial images taken at different times and from different angles, the accuracy and realism of the results may vary depending on the degree of differences between the images. Landmark detection provides a set of reference points, yet if corresponding points are not present in both images or if they differ significantly, alignment becomes problematic. Figure 26) illustrates this scenario with the left image captured six months before the right image. The middle image depicts the overlay result, highlighting the challenge of matching points and locating distinct points for creating a transformation matrix. To address this, we explored distance-based matching of landmark points, aiming to identify the best matching points based on their coordinates. However, variations in lighting conditions or other factors can lead to mismatches, underscoring the complexity of achieving accurate image overlay.

Figure 26) Overlay of images

Symmetry Check: Despite attempts to use a symmetry check approach, which considers differences between images, satisfactory results were not obtained. Variations in lighting conditions and inherent differences between facial sides hindered the effectiveness of this method. Figure 27) illustrates the challenges encountered in identifying symmetry in vitiligo-affected areas. Despite the potential for symmetry in vitiligo patterns, this approach did not provide reliable results, highlighting the need for alternative techniques in vitiligo detection.

Figure 27) Symmetry Check

Our endeavors to automate mask creation for vitiligo detection using traditional algorithms yielded limited success. The unique challenges posed by varying skin tones, lighting conditions, and inherent differences in facial features necessitated further exploration and refinement, ultimately leading us to leverage AI techniques for more robust and accurate mask generation.

Facial Vitiligo Research Outcome

The images are not specifically created for AI processing so the light conditions are not always ideal. When an image contains overexposure or the pigmented skin gets too much light, we get areas in the image that seem to be non-pigmented but are only overexposed. This difference is hard to tell. And in some cases, only well-educated clinical personnel can make a proper distinction.

Identification Challenges with High-Resolution Images: The use of high-resolution images provided detailed information but posed challenges in precisely defining vitiligo regions. Distinguishing between vitiligo-affected areas and overexposed regions due to flash lighting proved challenging, particularly for non-clinical personnel. This distinction is crucial for creating accurate AI models for automated detection.

Dynamic Region Analysis: Dynamic region analysis was employed to refine the accuracy of vitiligo region delineation. However, challenges persisted, and some results were inaccurate, highlighting the complexity of image processing, even for experienced medical professionals.

Adaptive Interface for Lighting Conditions: An adaptive interface was introduced to address varying lighting conditions, allowing for adjustments in lighting levels on both sides of the image. Despite its effectiveness in creating accurate masks, its clinical utility was limited due to complexity and time consumption.

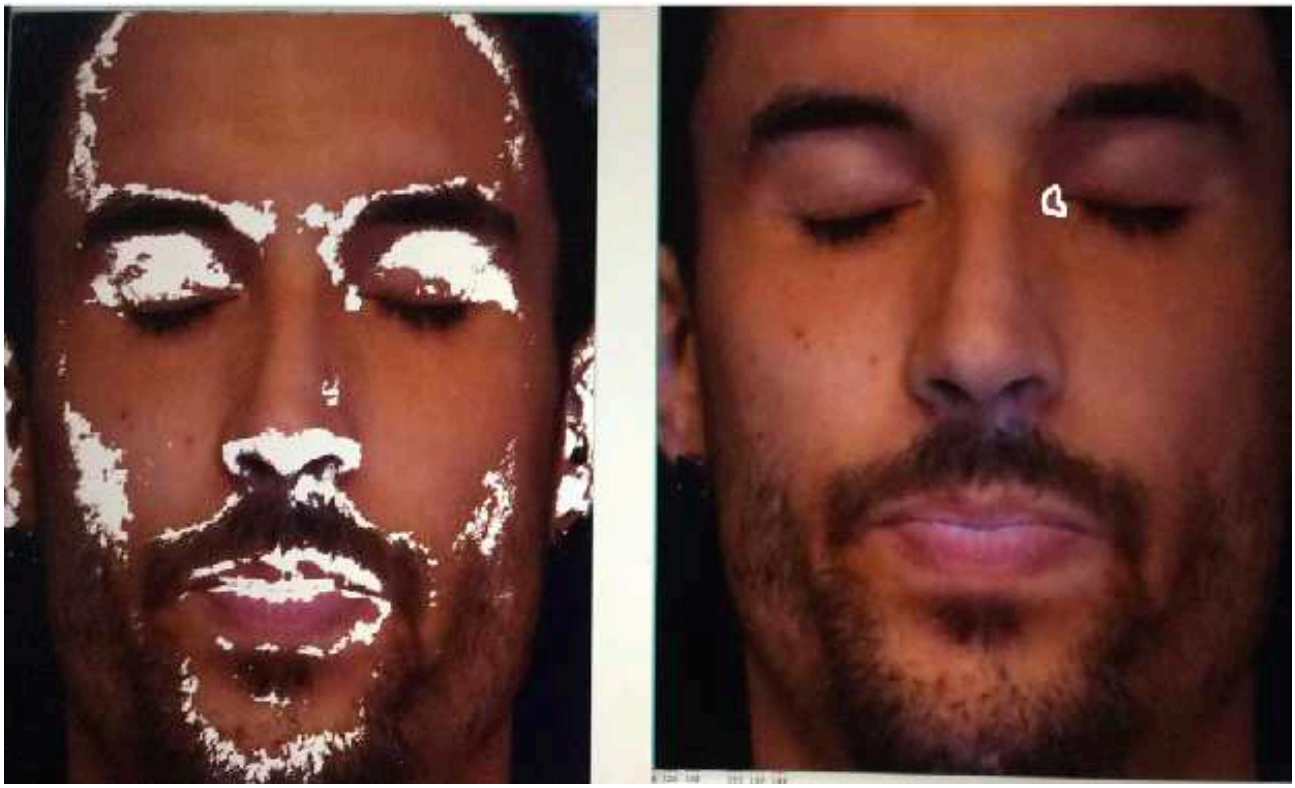
Challenges of Unilateral Lighting: Images captured with unilateral lighting presented unique challenges due to varying color temperatures on illuminated and shadowed sides of the skin. Potential solutions, such as the addition of a secondary light source for even lighting, were considered for future endeavors. As in the image provided, we have a selection with the same color range Figure 28) left vs on the right side the actual vitiligo region Figure 28) right. This shows that overexposure is difficult to distinguish from vitiligo.

Facial Landmarks and Image Splitting: Facial landmarks, particularly the nose point, offered a solution to address unilateral lighting by effectively splitting the image into two parts. This approach enhanced the precision of analysis by allowing separate processing of each image segment with tailored settings.

Improved Accuracy through Homogenization: Image homogenization allowed for separate processing of each image segment with tailored settings, enhancing the precision of analysis. Efforts were made to mitigate issues arising from varying light conditions and resulting shades, ensuring uniformity throughout.

Histogram Matching for Color Correction: Histogram matching emerged as a valuable tool in our research, facilitating color correction and constancy. By adjusting the histogram of an image to match that of a reference image, compelling results were achieved. This technique serves as a fundamental form of color correction without requiring complex machine-learning algorithms.

Figure 28) False Positive Detection



Homogenization Process: In our approach, we homogenize the image by determining the minimum and maximum values in the S Channel, which has improved our mask generation process. However, it's important to note that achieving perfect normalization remains challenging, especially for very dark or very light skin tones. In such cases, the homogenization process may introduce undesirable color shifts, posing a challenge in accurately identifying vitiligo in these regions.

Challenges and Future Directions: Despite advancements, challenges persist in dealing with variations in skin tones and lighting conditions, leading to false positives. Further improvements can be made to support the image processing quality and

accuracy. If overexposure and different light conditions can be eliminated this would be beneficial for AI processing.

Machine Learning and AI Algorithms

In our pursuit of automated vitiligo detection, we explored the application of machine learning and AI techniques. Leveraging existing face detection models, such as MediaPipe projects, provided access to extensive datasets trained on thousands of images with diverse features. However, we encountered specific challenges in adapting these models to vitiligo detection.

One key challenge was the limited availability of vitiligo-specific images within the comprehensive datasets, leading to misalignments and inaccuracies in model predictions. To address this limitation, we experimented with various strategies, including enlarging the region of interest and modifying the training data.

We trained a UNet architecture with adaptive masks created specifically for facial vitiligo detection. While the results showed promise, the network exhibited limitations in accurately identifying non-pigmented skin areas and delineating precise boundaries. Adjustments to the training data generation process were made to achieve a more balanced dataset and improve model performance.

Training the model with different datasets, including black and white images, RGB images, and variations in color channels, allowed us to evaluate performance under various conditions. After extensive training epochs, we observed significant improvements in accuracy across different image configurations.

We network with five different datasets. The dataset varied in type group through the data provided. We change the images to be:

1) Black and White
2) RGB Images
3) RGB Images with change channel (BGR) and added 15 pixels to the red channel
4) RGB Images and a sketch
5) RGB Images and a sketch overlay and Increased brightness and contrast

Training Results:

Computer time for this study was provided by the computing facilities of the MCIA (Mésocentre de Calcul Intensif Aquitaine)

1. Black and White image:

- After 100 epochs:
  7/7 [==========] - 28s 4s/sample - loss: 0.4094 - dice_coef: 0.5930
  - Accuracy: 59%
- After 600 epochs:
  7/7 [==========] - 27s 4s/sample - loss: 0.0394 - dice_coef: 0.9607
  - Accuracy: 96%

2. RGB images:
- After 100 epochs:
  7/7 [==========] - 3s 427ms/step - loss: 0.4167 - dice_coef: 0.5855
  - Accuracy: 58%
- After 600 epochs:
  7/7 [==========] - 3s 427ms/step - loss: 0.0399 - dice_coef: 0.9600
  - Accuracy: 96%

3. RGB images with changed channel and added 15 pixels to the red channel:
- After 100 epochs:
  7/7 [==========] - 3s 427ms/step - loss: 0.4099 - dice_coef: 0.5878
  - Accuracy: 58%
- After 600 epochs:
  7/7 [==========] - 3s 427ms/step - loss: 0.0392 - dice_coef: 0.9607
  - Accuracy: 96%

4. RGB images and a sketch:
- After 100 epochs:

  7/7 [==========] - 3s 423ms/step - loss: 0.3942 - dice_coef: 0.6057
  - Accuracy: 60%
- After 600 epochs:

  7/7 [==========] - 3s 422ms/step - loss: 0.0404 - dice_coef: 0.9596
  - Accuracy: 95%

5. RGB images and a sketch overlay and Increased brightness and contrast:
- After 100 epochs:

  7/7 [==========] - 3s 424ms/step - loss: 0.4263 - dice_coef: 0.5760
  - Accuracy: 57%
- After 600 epochs:

  7/7 [==========] - 3s 423ms/step - loss: 0.0554 - dice_coef: 0.9449
  - Accuracy: 94%

These training results provide insights into the performance of the different configurations. It is important to note that these numbers represent only a part of the overall network performance.

BW Images

Black and white images were utilized for training to enhance the contrast between non-pigmented and pigmented skin regions. The RGB images were converted into grayscale images, resulting in a single-channel representation.

Upon applying the trained model to process our images, it was discovered that there was no necessity to convert the images into black and white during inference. Instead, the patched data were directly fed into the trained model. Encouraging results were obtained for numerous images, demonstrating the model's ability to extract information across different patient positions and accurately identify regions with varying shades. However, the performance of the trained network was observed to be less satisfactory for certain skin tones. This discrepancy may be attributed to an insufficient amount of training data covering those specific skin types adequately. Furthermore, it was observed that the network's performance, trained solely with black and white (BW) images, was insufficient in terms of accuracy for the validation data. This limitation can be attributed to the loss of valuable color information during the conversion process from color to BW. Retaining the color information could provide crucial cues for the network to establish more accurate weights and enhance its segmentation capabilities.

RGB Color Images

The training of the data with RGB images in TIFF format resulted in significant progress after 1500 epochs. A high level of accuracy was achieved in predicting test images, demonstrating the effectiveness of the trained model. However, upon application to patient data, discrepancies between the predicted and actual vitiligo patterns were observed.

Incorporating color images into the training process introduced challenges stemming from variations in color temperature across different skin tones. Different skin tones exhibited varying levels of contrast differences, with darker skin tones typically displaying more pronounced contrast. Addressing this challenge requires the network to learn and adapt to the distinct characteristics of various skin colors. Alternatively, the utilization of separate networks dedicated to specific skin colors could facilitate specialization and enhance performance in each skin tone category.

RGB images with changed channel and added 15 pixels to the red channel

RGB images underwent processing to enhance the contrast between pigmented and non-pigmented skin regions. This involved adjusting the image channels by adding 15 pixels to the red channel and swapping the red and blue channels. As a result, the skin tone in the images appeared bluish, aiding in the differentiation of pigmented and non-pigmented areas. However, the prediction accuracy of test images using

this technique was found to be low, and application to patient data revealed significant discrepancies from the actual vitiligo patterns.

An investigation into the impact of training data size revealed that larger datasets did not necessarily lead to more efficient training. Surprisingly, the training set with only 60 images demonstrated better performance compared to a larger set of 1800 images.

Furthermore, the efficiency of the UNet architecture was influenced by the training process. It was observed that restarting the UNet multiple times on the same training data necessitated adjustments to the network architecture at the beginning of each session. This was evident from the initial drop in accuracy during training. Additionally, training the UNet for a large number of epochs without readjustment potentially affected its performance.

Comparative analysis indicated that training the UNet multiple times with shorter epochs resulted in improved performance. This adaptive approach allowed the network to refine its predictions based on the dataset. Notably, it was observed that while UNets can generate excellent results with a large number of epochs, training a single UNet for an extended period did not yield optimal accuracy.

The impact of training data quantity on training speed was evident, with networks trained with additional data showing accelerated training. A comparison between networks trained with 60 and 1800 images revealed that the former reached a similar state after 500 epochs as the latter did after 100 epochs. This underscores the importance of a sufficient and diverse dataset for achieving desirable results.

In conclusion, various techniques for image enhancement and feature enhancement were explored, including BW conversion, RGB channel manipulation, and sketch overlay. While promising outcomes were observed, particularly with RGB image processing, sketch overlay did not significantly enhance diagnostic accuracy compared to a standard RGB-trained network. The overhead of creating patches for sketch overlay did not justify the results obtained.

These findings suggest that an RGB-trained network holds more potential than employing sketch overlay for edge detection and feature enhancement in dermatological image analysis. Further research and refinement of these techniques are warranted to advance the diagnosis and treatment of vitiligo.

It's essential to note that while these techniques showed promise, further validation and evaluation with larger datasets and diverse patient populations are necessary to

establish their robustness and generalizability. By shedding light on the possibilities offered by image enhancement techniques, this study lays the groundwork for future research in computer-aided diagnosis and treatment of vitiligo.

Impact of Training Data Quality and Quantity:

The study investigated the impact of training data quality and quantity on the performance of the UNet architecture. Despite utilizing a limited amount of training data, ranging from 60 to 1800 images, and conducting multiple retraining sessions with epochs ranging from 1500 to 3400, the UNet architecture consistently delivered satisfactory results. The total number of epochs reached 66000 after multiple retraining sessions aimed at correcting mispredicted images and refining the network's performance.

An observation made during the study emphasized the critical role of training data quality in determining the network's performance. The accuracy and reliability of segmentation outcomes were heavily influenced by the precision and completeness of the ground truth data. Masks that inadequately covered non-pigmented areas or were overly wide posed challenges and compromised the segmentation accuracy.

Regarding training data quantity, it was found that a UNet achieved good accuracy with as few as 60 images trained for 1500 epochs. However, incorporating additional images into the training dataset notably improved the network's accuracy. The quality of the training data emerged as a crucial factor, as the network's performance relied on the diversity and representativeness of the provided images, encompassing various skin types, backgrounds, and pigmentation levels.

An illustrative example depicted in Figure 29 showcased the differences in training results between networks trained with varying approaches. The comparison highlighted noticeable variations in the prediction of vitiligo areas, with networks trained for longer durations with larger datasets exhibiting a nuanced understanding of active vitiligo regions. However, challenges arose in accurately distinguishing light-pigmented skin as vitiligo or not, particularly during the longer training processes.

Furthermore, the study revealed that retraining an existing network yielded lower accuracy compared to training a new network exclusively with new images. Even when additional epochs were applied to the existing network using new image data, the superiority of consistently training a new network was evident.

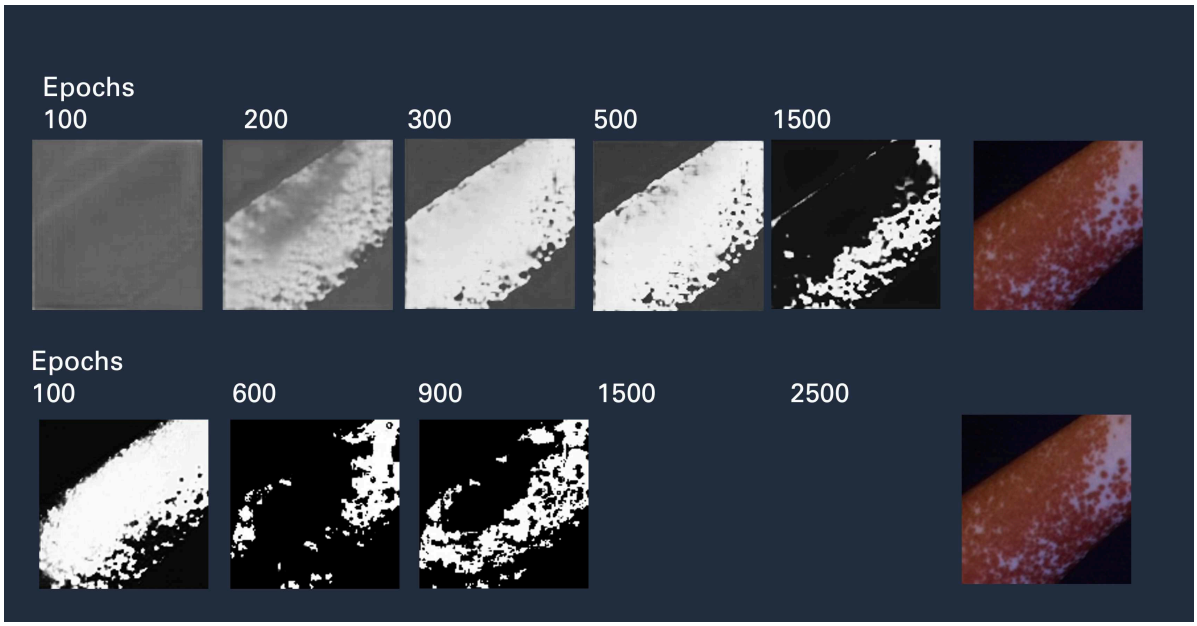
Continuous training, coupled with feedback and corrections, proved instrumental in refining the network's performance and enhancing its ability to accurately predict and classify vitiligo regions. The comparison between different training approaches

provided valuable insights into the trade-offs between data quality, training duration, and vitiligo detection accuracy.

Moreover, experiments involving patch downsizing from 256 to 128 did not yield improved performance. Downsizing the patches led to increased false positives, particularly in areas covering multiple skin types, highlighting the importance of maintaining patch size for precise predictions.

In conclusion, ensuring high-quality training data with precise and properly aligned masks is essential for optimizing the performance of the UNet architecture in vitiligo segmentation tasks.

Figure 29) Network Training Progress Visualisation

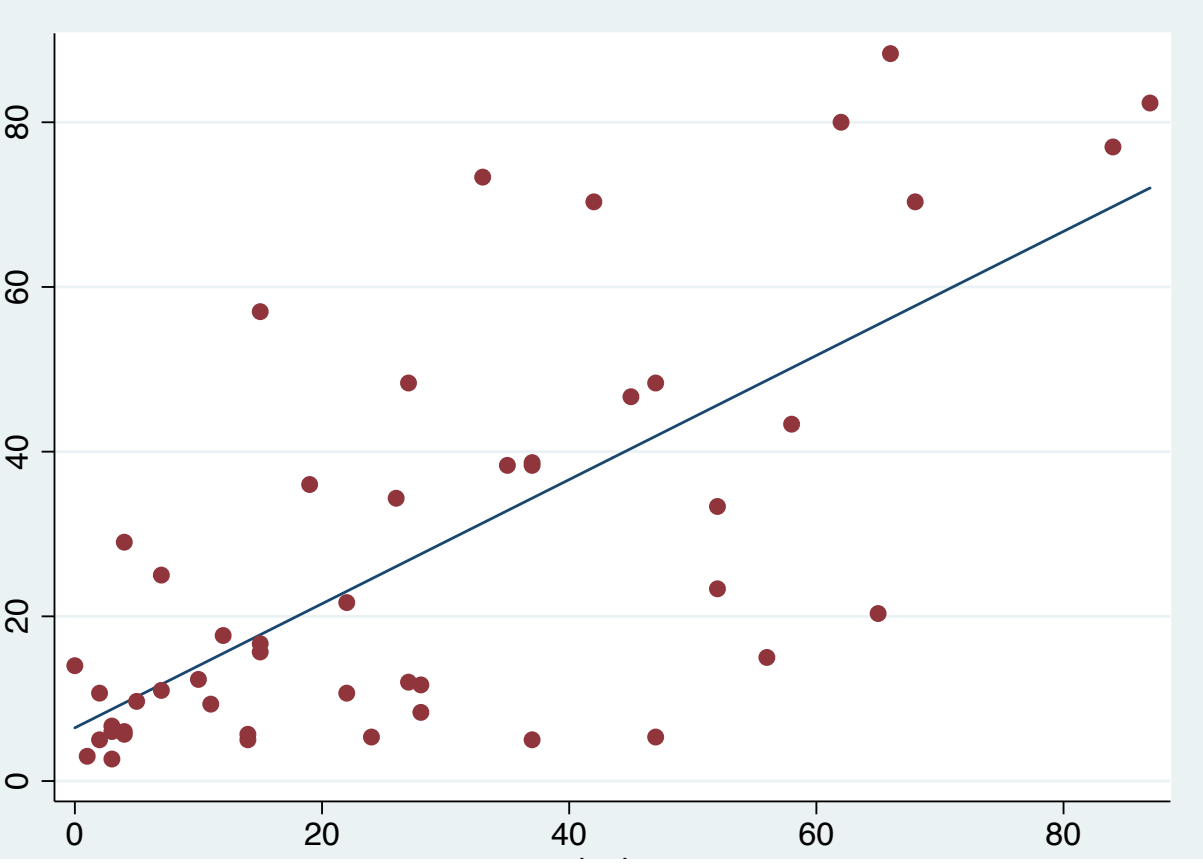


UNet and Deep Learning for Facial Images

The study compared the performance of an AI model with expert physicians in assessing facial vitiligo. Sixty-nine images of facial vitiligo from 59 patients were analyzed, and three expert physicians provided assessments of the vitiligo-affected facial surface area. Overall inter-rater and intra-rater reliability were excellent, and correlation analysis between physician scores and AI predictions showed good data clustering around the regression line. However, differences were noted, especially for patients with varying surface areas of the face involved, potentially influenced by factors such as image acquisition and human variability.

Three expert physicians assessed the vitiligo-affected facial surface area at two-time points. The surface of the vitiligo area was scored from 0% (no vitiligo) to 100% (complete depigmentation of the area). As shown in Table 2, the overall inter-rater and intra-rater reliability was excellent with ICCs of 0.80 (95% CI:0.71-0.80) and 0.81 (95% CI: 0.78-0.90), respectively. Correlation analysis between the mean score of all physicians and the AI showed good data clustering around the regression line of the scatter plot (Figure 30).

Figure 30) Regression Line



However, AI values were not always directly comparable with the expert physicians' assessment, especially for patients with a small or large surface area of the face involved. These differences could be due to the image acquisition, or a tendency for human deviation based on experience and the global estimation of the area in question (Figure 31a and b).

Figure 31a) Patient Results

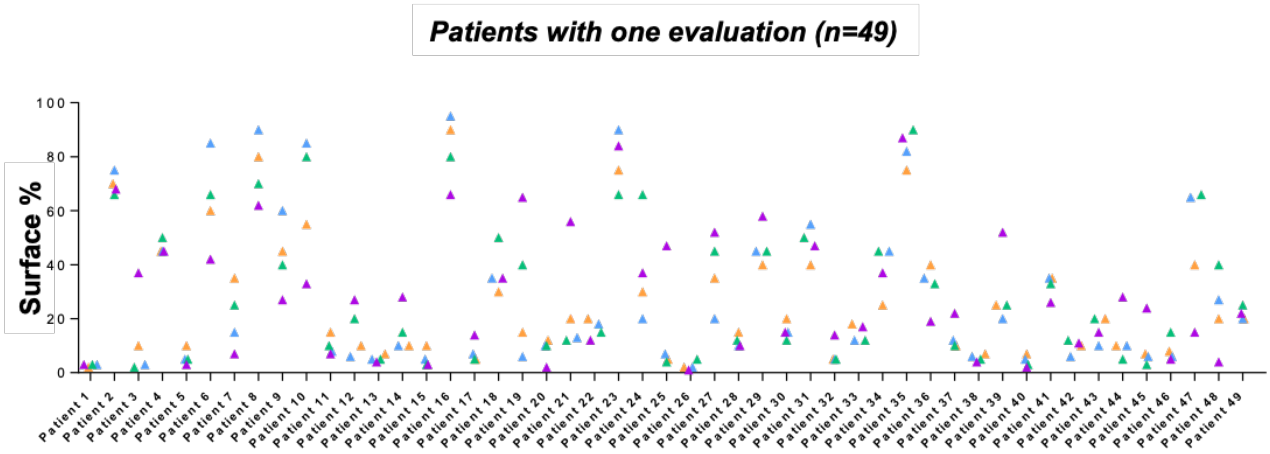
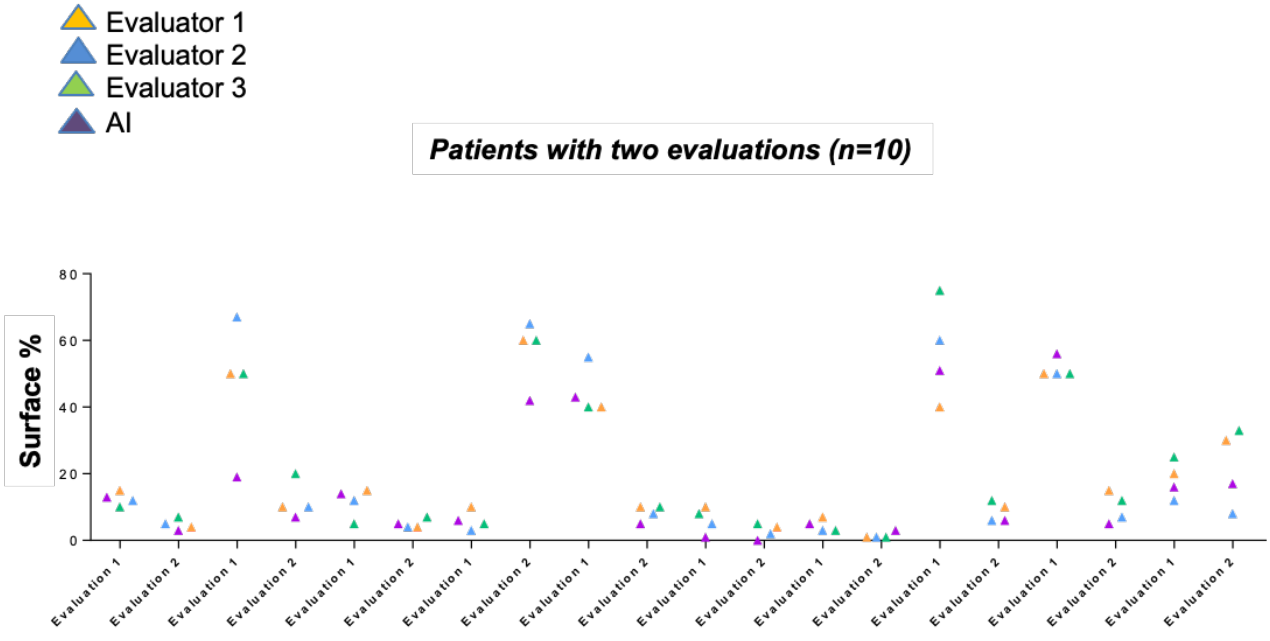


Figure 31b) Patient Results



Challenges Related to Image Backgrounds and Lighting

The study encountered challenges related to image backgrounds and lighting conditions during image capture. These challenges included variations in background patterns, shadowing effects, and overexposure, which could impact the accuracy of vitiligo analysis. Strategies to overcome these challenges involved adjustments to image processing techniques and improvements in lighting setups.

Overexposure and Positioning Challenges

Additional lighting was employed during image capture, resulting in the formation of shadows on one side of the subject. This shadowing introduced variations in pixel values, further complicating the image analysis process. Another issue arising from the lighting setup was overexposure. Areas of the skin that received excessive light appeared bright white, which could be easily misinterpreted as vitiligo, adding another layer of difficulty to the analysis.

Methods for Overcoming Challenges

To address the challenges posed by image backgrounds and lighting conditions, various computer graphics methods were employed, including erosion and dilation techniques and adjustments to lighting setups. Additionally, a UNet-based approach was implemented to remove hair and beard regions from images, enhancing the quality of masks and aiding in more accurate vitiligo detection.

Outcome of Body Image Processing in Vitiligo Research

Training the network to accurately detect vitiligo on various skin tones posed a significant challenge. The subtle differences between non-pigmented and pigmented skin areas often amounted to just a few pixel values, making precise differentiation difficult.

Our exploration of dynamic mask generation and template matching techniques for object detection provided valuable insights into their strengths and limitations. While these methods offer promise, they also present challenges, particularly in handling the complexities and variations encountered in real-world scenarios. This understanding serves as a foundation for the development of more sophisticated and adaptive object detection techniques tailored to the specific needs of vitiligo research.

Given our focus on the head and hands, it was imperative to develop a method for accurately locating the face within the image. The adaptive mask generation approach, described previously, proved instrumental in generating high-quality masks conducive to training UNets for facial detection.

Body pictures

Expanding the application of the trained network for facial vitiligo to body images revealed several challenges unique to this context. The differing lighting conditions and skin tones between the head and lower body necessitated the development of a separate UNet architecture tailored specifically for body images. While the UNet architecture utilized for the face provided a solid foundation, it required adjustments to effectively address the complexities of body image analysis.

One significant challenge encountered was the presence of a diverse range of skin information, particularly on the chest. The chest area often contains non-vitiligo regions with no skin pigmentation, especially in women who may not have been exposed to sunlight. This variability in skin tone and texture presented difficulties in accurately detecting vitiligo regions.

In practical use, analysis of body images demonstrated that while accurate evaluation of vitiligo regions is achievable in a limited context such as the head, challenges arise when extending the analysis to the full body. Images containing natural non-pigmented skin alongside non-pigmented vitiligo skin introduced limitations on AI capabilities. Overcoming these limitations requires the development of networks capable of distinguishing between normal and vitiligo-affected skin across varying skin tones.

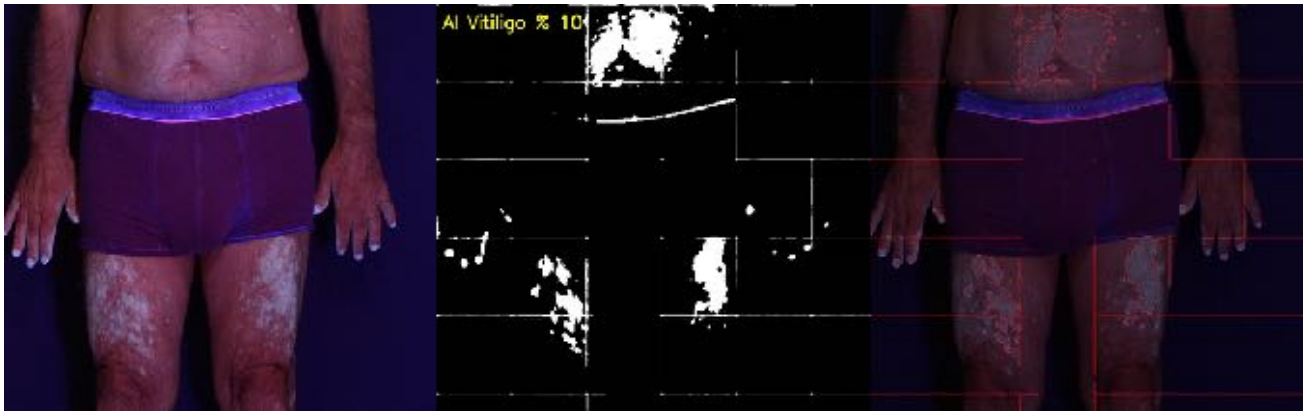
Figure 33) Non-Pigmented Skin vs Vitiligo

A) Original Image B) Original Image with Vitiligo Mask

Additionally, broader views of the complete body introduced further complexities, including the presence of different skin tones within a single image and the use of white underwear and floors. White underwear and floors could create false positive detections and impact segmentation accuracy by misleading the network to associate white regions with vitiligo.

To address these challenges, meticulous selection and preprocessing of training data were crucial. Strategies included using images without white underwear Figure 32), manually editing images to remove underwear, and augmenting data with rotation, scaling, and random cropping. Furthermore, replacing white floors with a neutral color during manual editing and normalizing intensity levels to mitigate lighting variations were essential steps to improve segmentation accuracy.

Figure 32) Body Image Analysis



Additional challenges included variations in arm positions and identifying normal skin regions in the absence of reference skin. Techniques such as adjusting lighting, employing specialized algorithms for shadow detection, and leveraging context and texture analysis were employed to overcome these challenges.

The challenge is further compounded by the need to accurately locate vitiligo regions on pigmented skin, where overlapping areas present additional complexities. Addressing these challenges requires continued refinement of network architectures and preprocessing techniques to enhance segmentation accuracy and robustness in clinical settings.

In conclusion, the analysis of body images for vitiligo research presents multifaceted challenges related to white underwear, flooring, lighting variations, arm positions, and identifying normal skin regions. Careful selection and preprocessing of training data, along with augmentation techniques and advanced algorithms, are essential for overcoming these challenges and improving segmentation accuracy in clinical practice.

Image Preparation

While neural networks can achieve impressive results, they are not infallible and require careful preprocessing of input data to optimize performance. In the case of body images for vitiligo analysis, preprocessing involves filtering out unwanted elements such as white underwear, hair, and the grey PVC floor.

White underwear presents a particularly challenging issue, as its color values may overlap with those of non-pigmented skin areas. To address this, various techniques can be employed. One approach involves applying a filter based on the HSV color space, where the maximum value (V) is set to a threshold below the white underwear value. However, this method risks removing both the underwear and portions of the skin, necessitating careful adjustment to balance the removal of unwanted elements while preserving skin areas.

Erosion and dilation operations can also aid in refining the mask generated from the filtered image. Erosion reduces the size of the mask, effectively separating smaller objects from larger ones, which can help remove small contours and noise. Dilation then expands the mask, restoring lost areas and ensuring that the remaining contours accurately represent the body.

Iterative refinement may be necessary during this process, as overly aggressive filtering could inadvertently remove portions of the skin. Adjustments to the erosion and dilation parameters may be required to strike the right balance between noise removal and preservation of skin regions.

Once the initial preprocessing steps are complete, the resulting mask should ideally encompass the entire skin area while minimizing noise and unwanted elements. However, the presence of the grey PVC floor poses a unique challenge, as it is more prominent in the image and cannot be easily filtered out using standard techniques. In such cases, cropping the lower portion of the image containing the floor may be necessary to ensure accurate segmentation. If those steps are no success the only way to overcome this issue is by replacing the floor. It needs to be a different color that can not be mixed with vitiligo skin color.

Overall, the preprocessing of body images for vitiligo analysis is a complex and iterative process, requiring careful consideration of various factors such as underwear removal, noise reduction, and floor cropping. While similar techniques may have been applied to head images, the challenges presented by body images are more complex due to the presence of multiple skin tones and the prominence of the floor. Iterative refinement and adjustment of preprocessing techniques are essential to ensure accurate segmentation and analysis of body images in vitiligo research.

Challenges of Vitiligo Prediction

When analyzing the results of UNet processing on images, it becomes evident that the network performs admirably on images with good contrast. However, challenges arise when dealing with light skin tones, where the subtle differences between pigmented and non-pigmented areas may not be adequately recognized by the AI network. This issue becomes particularly pronounced in regions such as the chest or legs, where extremely light skin tones can resemble vitiligo, leading to misclassification by the network.

To address this challenge and enhance contrast, techniques such as Histogram Matching can be employed. By transferring color information from images where non-pigmented skin is identifiable to other images, the color temperature can be aligned, resulting in darker and redder tones indicative of pigmentation. This adjustment allows the network to more accurately differentiate between pigmented and non-pigmented areas.

The versatility of AI networks allows for the training of models tailored to specific tasks, offering a promising avenue for addressing challenges in vitiligo prediction. Unlike traditional computer graphics techniques, AI networks can identify skin sections that may be difficult to locate using conventional methods such as HSV settings.

However, significant challenges persist, particularly concerning variations in lighting conditions and skin colors. Training networks to accommodate this diversity while ensuring robust performance remains a daunting task. Approaches such as normalization and destabilization have been explored, with normalization showing promising results in terms of training data accuracy.

For instance, training an Attention Res UNet network on a dataset comprising 690 patches with a dice coefficient yielded impressive results after 3400 epochs, with a loss of 0.0036 and an accuracy of approximately 99.64%. These numbers underscore the high standard achieved through rigorous training. The training data

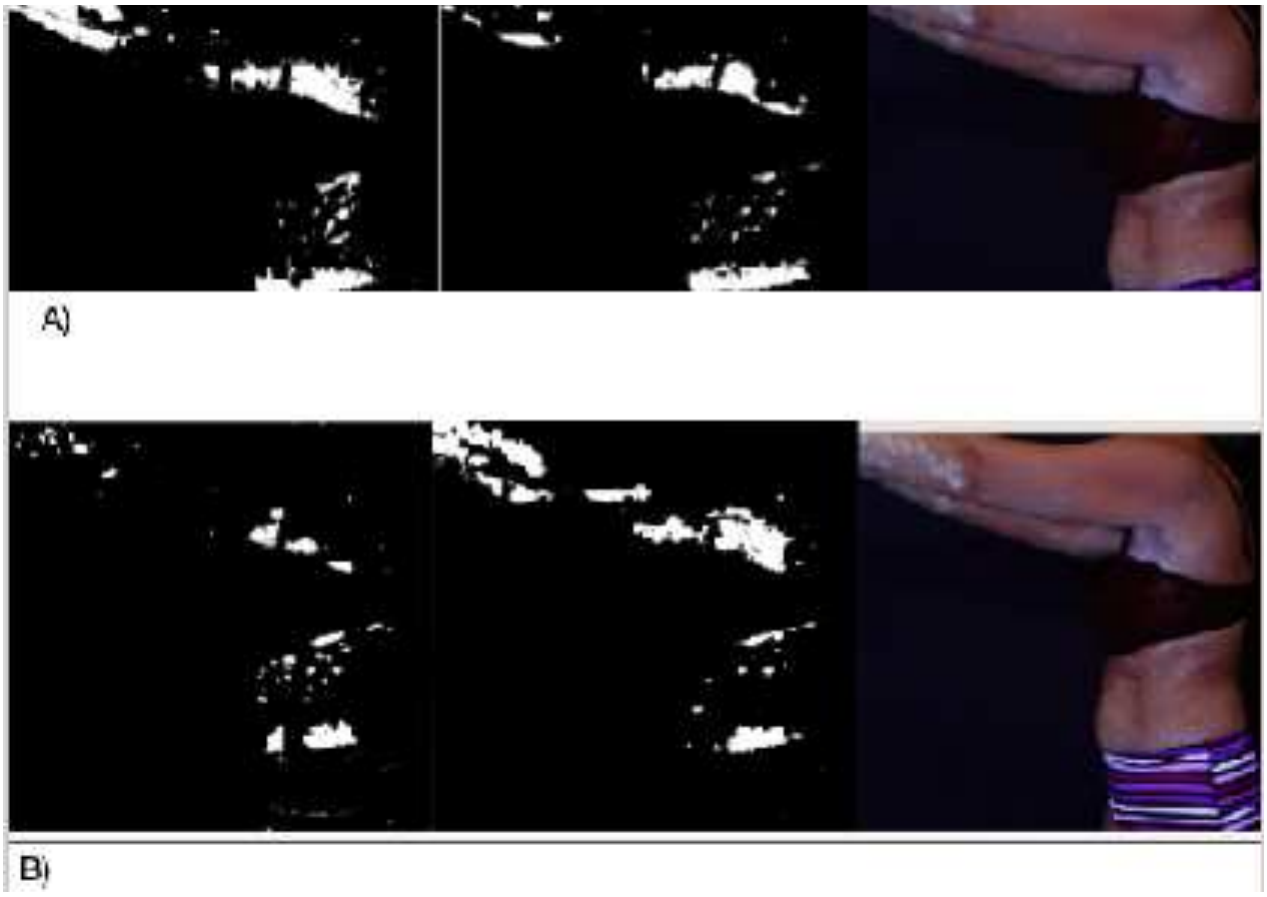
consisted of unmodified 128x128 patches, normalized and trained to optimize performance.

Epoch 3400/3400
7/7 [==============================] - 3s 476ms/step - loss: **0.0035** - dice_coef: **0.9965**

Visualizing the training progress through image validations at intervals such as 100 epochs offers valuable insights into network performance. However, it's crucial to recognize that while retraining can enhance performance, ongoing optimization, and refinement are necessary to maintain accuracy over time. The results obtained from retrained networks highlight their potential but also emphasize the importance of continuous optimization to adapt to evolving challenges and datasets.

The results of the networks show the training results after training for Figure 34A) 3500 and after training for 7000 epochs Figure 34B)

Figure 34) Network Training Comparison



A) Training results after 3500 Epochs B) Training Results after 7000 Epochs

If we compare the networks on a sample test image we get the following results in Figure 35A) we have a 256 patch analyzed and in Figure 35B) we see Patch 128 the original image is shown in Figure 35C). The results show that the larger area provided to the network is not improving the correct detection of the network. It also shows that even though the 128 patch size is more accurate it is not getting all vitiligo the darker regions show a lack of accuracy. Here we need to improve the training with darker skin or skin more in the shade and lighter skin. The networks are not that flexible to work accurately in different light conditions.

Figure 35) UNet comparison 256 vs 128 Patch Size

A) UNet 256 patch size B) UNet 128 patch size C) Original image

Caption

AI Processing Outcome

The quality and diversity of training data significantly influence the performance of the UNet network. Accuracy in labeling and diversity in the dataset is essential for creating a robust training dataset. A well-balanced mix of background, pigmented skin, and vitiligo patches is crucial for effective network training. However, we observed that training data collected from one body part did not generalize well to others, emphasizing the need for region-specific training. Variations in lighting conditions also posed challenges in accurately recognizing segments. Retraining affected segments under different lighting conditions was necessary to improve

network accuracy and ensure consistent performance across diverse scenarios. Therefore, continual refinement and adaptation of the training data are essential to address these challenges and optimize segmentation performance.

Insufficient training data can lead to overfitting, where the network fails to generalize well to new images and performs poorly on unseen data. This phenomenon was observed when using a small dataset, resulting in the network memorizing the training data rather than learning meaningful features. High numbers of epochs exacerbate overfitting, as the network becomes overly tuned to the training data. Overfitting can be mitigated by increasing the diversity of the training data and applying data augmentation techniques such as morphing and rotation. Augmenting the dataset with diverse examples helps the network learn robust features and reduces the likelihood of overfitting, ultimately improving segmentation performance on unseen images.

Achieving a balanced dataset is crucial for network robustness. In addition to including pigmented and non-pigmented skin patches, incorporating a diverse range of skin tones is essential for comprehensive training. Challenges arose when processing images with varying skin tones within the same image, as the network struggled to identify light skin tones with vitiligo accurately. This limitation stemmed from an insufficient representation of non-pigmented skin in our training set. Creating separate datasets for specific body parts can be a preferable strategy, allowing for targeted training and better segmentation performance.

Despite efforts to mitigate challenges related to lighting conditions and skin color variations, significant obstacles persisted. Methodologies such as partitioning images and utilizing separate networks for distinct sections or skin types were explored. However, their effectiveness was limited due to the lack of suitable images specifically tailored for AI processing. Suggestions to improve image quality, such as incorporating additional lighting sources during image capture, were proposed but not fully resolved due to the dataset primarily consisting of older images not optimized for AI applications. This highlights the need for dedicated image acquisition protocols tailored to AI requirements.

Optimizing the training process is pivotal for improving the efficiency and efficacy of AI models. Our approach involved several strategies aimed at refining the training process, including retraining models with newly annotated data to enhance segmentation accuracy. Visualizing training progress through image validations provided valuable insights into network performance. However, the effectiveness of complete retraining versus incremental data addition varied, emphasizing the

complexity of optimization strategies and the importance of comprehensive training approaches.

AI offers a valuable tool for streamlining traditional clinical assessments, reducing variability in diagnoses and treatment planning. Despite potential flaws, AI provides valuable insights, significantly reducing manual workload in image analysis and evaluation. By automatically generating reports and identifying affected areas, AI facilitates rapid assessment, allowing clinicians to focus on refining and validating results. The seamless integration of AI into the clinical workflow enhances efficiency and productivity, with the entire process completed in less than a minute.

Several promising avenues warrant exploration for further research and development in AI-based medical image analysis. Innovations such as projecting skin onto a 3D puppet model for dynamic visualization and scaling up training datasets with higher image resolutions hold the potential for addressing existing challenges. Training networks on complete body parts rather than isolated regions could enhance segmentation accuracy and clinical utility. By advancing AI capabilities, we can improve diagnosis, treatment planning, and patient care in dermatological conditions like vitiligo.

AI Model Training and Evaluations

One UNet network was trained with different datasets.

Training Results:

Computer time for this study was provided by the computing facilities of the MCIA (Mésocentre de Calcul Intensif Aquitaine)

The results we get for the dice_coef model are presented here:

1. Black and White image:
   - After 100 epochs:
     7/7 [==========] - 28s 4s/sample - loss: 0.4094 - dice_coef: 0.5930
     - Accuracy: 59%
   - After 600 epochs:
     7/7 [==========] - 27s 4s/sample - loss: 0.0394 - dice_coef: 0.9607
     - Accuracy: 96%
2. RGB images:
   - After 100 epochs:
     7/7 [==========] - 3s 427ms/step - loss: 0.4167 - dice_coef: 0.5855
     - Accuracy: 58%
   - After 600 epochs:
     7/7 [==========] - 3s 427ms/step - loss: 0.0399 - dice_coef: 0.9600
     - Accuracy: 96%

3. RGB images with changed channel and added 15 pixels to the red channel:
   - After 100 epochs:
     7/7 [==========] - 3s 427ms/step - loss: 0.4099 - dice_coef: 0.5878

- Accuracy: 58%
  - After 600 epochs:
    7/7 [==========] - 3s 427ms/step - loss: 0.0392 - dice_coef: 0.9607
    - Accuracy: 96%
4. RGB images and a sketch:
  - After 100 epochs:

    7/7 [==========] - 3s 423ms/step - loss: 0.3942 - dice_coef: 0.6057
    - Accuracy: 60%
  - After 600 epochs:

    7/7 [==========] - 3s 422ms/step - loss: 0.0404 - dice_coef: 0.9596
    - Accuracy: 95%
5. RGB images and a sketch overlay and Increased brightness and contrast:
  - After 100 epochs:

    7/7 [==========] - 3s 424ms/step - loss: 0.4263 - dice_coef: 0.5760
    - Accuracy: 57%
  - After 600 epochs:

    7/7 [==========] - 3s 423ms/step - loss: 0.0554 - dice_coef: 0.9449
    - Accuracy: 94%

The trained networks demonstrated high accuracy on test and validation data, indicating successful completion of the training process. However, the ultimate test lies in applying these networks to full-body pictures encompassing various lighting conditions.

Running the network on 274 full-body images revealed approximately 120 inaccurately predicted images. The main issues identified include:

White Underwear Detection: The network misidentified white underwear as vitiligo patches. To mitigate this, we implemented a method to remove white underwear pixels before applying the AI network.

Hair on patients caused false positive results. This issue was addressed by incorporating hair-containing patches into the training set and developing a dedicated network for hair detection.

Missed Vitiligo Regions: Some vitiligo regions, particularly darker areas differing slightly in tone, were not accurately identified. To improve network accuracy, additional patches representing these regions were added to the training data.

Several strategies were employed to address these challenges, including the Dilate and Erode technique for removing white underwear, dedicated networks for hair detection, and augmentation of the training data to include diverse skin tones.
To continually enhance the accuracy of the U-net model, a periodic training approach was implemented. This involves automatically splitting generated images and masks, loading the model, and conducting new training runs to further improve performance.
In conclusion, while challenges were encountered during testing, the iterative training process and implementation of remediation strategies have contributed to the continuous enhancement of the U-net model's accuracy. These findings underscore the importance of ongoing refinement and adaptation in the development of AI-based medical image analysis systems.

Body Image AI Outcome

In our efforts to optimize patch data for training the network, we explored two distinct approaches, each aimed at improving the representation and quality of patches to enhance the network's performance.

Augmentation with Additional Features:
The first approach involved augmenting patches with additional features, specifically sketch information. By incorporating sketch data, we aimed to provide the network with richer and more diverse information to facilitate better segmentation accuracy. This augmentation process was iteratively applied during training, leading to promising results.

After 100 epochs, the network achieved an accuracy of approximately 69%, steadily improving over subsequent epochs. Ultimately, after 700 epochs, the accuracy reached an impressive 95%, highlighting the effectiveness of augmenting patches with sketch information.

Histogram Alignment of Images:
The second approach focused on the histogram alignment of images to enhance color intensity across patches. This technique aimed to improve the network's ability to capture subtle variations in skin tone and texture. Training the network with these modified patches resulted in a remarkable accuracy of 98% after 3500 epochs, demonstrating the efficacy of this modification technique.

Despite the promising results obtained with these augmentation approaches, further evaluation of new images revealed unexpected findings. Networks trained with augmented patches struggled to generalize well to unseen data, indicating that excessive augmentation introduced complexity hindering accurate segmentation of new images.

In contrast, networks trained with non-modified image data consistently outperformed augmented variations, underscoring the importance of maintaining simplicity in training data.

To refine the dataset, we used a slider program to create patches of size 128x128, ensuring a balanced mix of pigmented, non-pigmented, and background patches. This iterative approach allowed us to enrich the model by subsequently adding more training data, leading to improved accuracy with each iteration. GPU acceleration significantly reduced training time, and experimenting with different cost functions revealed that the Dice coefficient outperformed the Jaccard index in terms of network predictions.

The choice of patch size significantly impacted the quality of output masks, with further quantitative analysis needed to comprehensively assess performance differences. Processing time per image with the network was approximately 2 minutes, with potential increases when using smaller patches.

Overall, these approaches and refinements highlight the ongoing efforts to optimize AI-based segmentation models for accurate and efficient analysis of body images in vitiligo diagnosis and treatment planning.

Data Loading and Preprocessing:

Image loading was performed using the OpenCV read () method, enabling the reading of image data in various formats.

- To ensure proper data normalization, the image data was converted into a float32 array.
- Alternatively, TensorFlow's image-loading methods were utilized, allowing for immediate data normalization without additional conversions.

Limitations and Challenges for Body Image Concatenations

The necessity to concatenate full-body images for vitiligo calculation highlighted several challenges, underscoring the complexity of achieving seamless image alignment.

Missing or Incorrect Key Points: One significant challenge arose during the implementation of the concatenation algorithm when inadequate overlap between images resulted in the inability to identify valid key points for accurate matching. This issue was particularly pronounced when combining upper body regions with images containing feet, where the absence of suitable key points impeded precise alignment.

Patient or Camera Movement: Challenges stemming from patient or camera movement were encountered, leading to subtle shifts in image positions. Even slight movements during image capture posed difficulties in achieving precise alignment between images.

Different Positions Yield Different Connection Points: Variations in body positions introduced additional complexity, with distinct connection points observed for different body poses. Aligning images featuring extended arms, for example, required different matching strategies compared to images with the arms positioned closer to the body.

Insufficient Overlap in Images: In some instances, images lacked sufficient overlap, rendering traditional template or border-matching techniques ineffective. This limitation necessitated alternative approaches to achieve successful concatenation.

Floor Color: The similarity in color between the floor and non-pigmented skin posed a significant challenge, as existing removal techniques were unsuccessful in distinguishing between the two. Addressing this issue is crucial to ensure accurate vitiligo detection by AI networks.

The most effective approach employed a matching template generated from a lower image patch, consisting of 100 lines, which was then shifted across the upper image. This method facilitated successful image concatenation, as depicted in Figure 35-1).

Figure 35-1) Image Concatenation

These challenges underscore the intricacies of image concatenation and emphasize the importance of robust solutions to address key point detection, patient or camera movement, varying body positions, insufficient overlap, and floor color discrepancies. Additionally, ensuring a complete image set is available is essential to avoid misconnections in the concatenation process. Future improvements may involve implementing object references in the background to facilitate more accurate concatenation and enhance overall results.

The most effective approach we implemented involves using a matching template generated from a patch of the lower image, consisting of 100 lines, which is then shifted from the top to the bottom of the upper image. Figure 35-1) demonstrates the concatenation of two images using this method.

Online System for Vitiligo

The development of an online system for image upload, storage, and analysis was initially envisioned as a pivotal component of our research methodology, facilitating efficient data collection and management from various sources. However, our plans encountered a hurdle due to constraints within the university infrastructure. Consequently, we were unable to deploy the system on university hardware as intended, and instead, had to rely on a local system.

This limitation had significant implications for our research. Primarily, it affected the scale and diversity of the image dataset available for analysis. A comprehensive dataset is crucial for training robust machine learning models and gaining accurate insights. The absence of a public online system constrained our ability to expand the dataset to its full potential.

Moreover, the unavailability of the online system hindered effective collaboration with external partners. A centralized platform would have facilitated seamless data sharing and collaboration, potentially enhancing the scope and depth of our research endeavors.

In response to these resource constraints, we adapted our research approach, focusing on optimizing the analysis of the available dataset and exploring alternative avenues for collaboration and data acquisition. It's worth noting that collaboration with other clinics, particularly those equipped with UV light facilities, became essential due to the limitations imposed by the absence of the online system. Although the online system could not be fully implemented, we developed a software base capable of dynamic retraining, which would have allowed for on-the-fly updates instead of batch processing and FTP transfers to remote servers. While this

functionality could not be realized in practice, the groundwork laid for the software development remains valuable for future research endeavors.
In conclusion, the absence of the online system highlighted the importance of resource allocation and meticulous planning in research initiatives, prompting us to adapt and innovate within the constraints imposed by the infrastructure limitations.

Conclusion and Take-Home Message

With the development of AI networks, we have demonstrated the ability to distinguish between pigmented and non-pigmented skin. Challenges such as floor color, lighting conditions, and white underwear can be mitigated through specific clinical pre-settings. The trained AI networks are effective across a wide variety of skin types; however, continuous retraining and extension of these networks remain crucial. Additionally, creating specialized networks for specific body parts is essential for improving accuracy and performance.

The introduction of an online patient system opens the door for a broader reception of new patients and enhanced vitiligo analysis. While the system needs to be made more secure, the foundational infrastructure for communication is already in place. Furthermore, the use of mobile phones can be extended and connected to the platform, enhancing accessibility and usability.

Integrating this technology into the clinical workflow is the next step. Automating the vitiligo report generation process will streamline clinical operations, allowing for faster and more accurate assessments. This integration will ultimately lead to better patient outcomes and more efficient management of vitiligo.

The nature of vitiligo and the fact that skin can pigment again highlight that AI cannot fully replace skilled human judgment. While AI can be trained to distinguish between pigmented and non-pigmented skin, this distinction remains very nuanced and complex.

Thanks

My deepest gratitude to Martin Hagedorn for his invaluable guidance, mentorship, and unwavering support throughout the entire thesis process. His expertise and insights have been instrumental in shaping the direction of this research and in helping me navigate through its complexities.

I would also like to express my sincere appreciation to the members of the Thesis Jury for their time, expertise, and thoughtful feedback during the evaluation of this work. Your insights and constructive criticism have contributed significantly to the refinement and improvement of this thesis.

Julien Senechal: Thank you for generously providing materials and sharing your expertise, which greatly enriched my research.

Nils Purschke: Your valuable insights into the UNet were deeply appreciated.

Slim Karkar: I am grateful to you for granting access to the CURTA Cluster, which facilitated GPU access and accelerated the computational aspects of our research.

Anthelme Pradeau: Special thanks for your outstanding work in creating the mobile interface, which greatly enhanced the usability and accessibility of our project.

And I am very grateful for the support of my family and friends Stéphanie La Merle and Alain Frey.

References

[1] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639), 115-118.

[2] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. Medical image analysis, 42, 60-88.

[3] Ronneberger, O., Fischer, P., & Brox, T. (2015). UNet: Convolutional networks for biomedical image segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 234-241). Springer, Cham.

[4] Chen, H., Zhang, Y., Zhang, W., Liao, P., & Li, K. (2018). Deep learning-based classification of skin tumors using dermoscopy images. BMC Bioinformatics, 19(1), 1-13.

[5] Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., ... & Webster, D. R. (2016). Development and validation of a deep learning algorithm for the detection of diabetic retinopathy in retinal fundus photographs. JAMA, 316(22), 2402-2410.

[6] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

[7] Litjens, G., Ciompi, F., Wolterink, J. M., de Vos, B. D., Leiner, T., Teuwen, J., ... & Išgum, I. (2017). State-of-the-art deep learning in cardiovascular image analysis. JACC: Cardiovascular Imaging, 10(3), 257-270.

[8] Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K., ... & Ng, P. (2019). A guide to deep learning in healthcare. Nature Medicine, 25(1), 24-29.

[9] Albouy, A., et al. (2020). "Imaging the Dynamics of Melanocytes in Xenopus laevis Tadpoles In Vivo." Journal of Visualized Experiments, (159), e61247.

[10] Dunn, K. W., et al. (2011). "Confocal imaging in biology and medicine." New York: Springer Science & Business Media.

[11] Li, Y., et al. (2019). "Quantitative analysis of skin pigmentation using image processing techniques." IEEE Transactions on Biomedical Engineering, 66(8), 2244-2253.

[12] Low, T. Y., et al. (2017). "Deep learning for melanoma detection: From traditional models to modern deep learning." In Computer Vision - ECCV 2016 Workshops (pp. 250-263). Springer International Publishing.

[13] Madabhushi, A., et al. (2006). "Image analysis and machine learning in digital pathology: Challenges and opportunities." Medical Image Analysis, 10(4), 358-369.

[14] Pantoja, C., et al. (2019). "Xenopus as a Model System for Studying Eye Development and Disease." Seminars in Cell & Developmental Biology, 98, 193-204.

[15] Ronneberger, O., et al. (2015). "UNet: Convolutional networks for biomedical image segmentation." In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 234-241). Springer International Publishing.

[16] Sahu, S., et al. (2020). "Automated analysis of skin melanoma: Advancements in imaging, segmentation, and diagnosis." Biomedical Signal Processing and Control, 60, 101984.

[17] Stork, D. G., et al. (2017). "Systematic Biology for Basic and Applied Research." Systematic Biology, 66(1), e1-e2.

[18] Waghule, T., et al. (2018). "ImageJ Plugin for Label-Free Estimation of Skin Melanoma Thickness." Journal of Medical Engineering & Technology, 42(8), 619-628.

[19] Abràmoff, M. D., et al. (2016). "A Clinically Applicable Approach to Continuous Prediction of Future Risk of Progression to Advanced Age-Related Macular Degeneration." JAMA Ophthalmology, 134(12), 1447-1455.

[20] Burlina, P. M., et al. (2017). "Automated Grading of Age-Related Macular Degeneration From Color Fundus Images Using Deep Convolutional Neural Networks." JAMA Ophthalmology, 135(11), 1170-1176.

[21] Chakravarthy, U., et al. (2014). "Clinical risk factors for age-related macular degeneration: a systematic review and meta-analysis." BMC Ophthalmology, 14(1), 1-16.

[22] Fraz, M. M., et al. (2012). "An approach to localize the retinal blood vessels using bit planes and centerline detection." Computer Methods and Programs in Biomedicine, 108(2), 600-616.

[23] Grassmann, F., et al. (2017). "Genetics of age-related macular degeneration." Progress in Retinal and Eye Research, 57, 64-106.

[24] Gulshan, V., et al. (2016). "Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs." JAMA, 316(22), 2402-2410.

[25] Sim, D. A., et al. (2017). "Automated Retinal Image Analysis for Diabetic Retinopathy in Telemedicine." Current Diabetes Reports, 17(10), 1-10.

[26] Staal, J., et al. (2004). "Ridge-based vessel segmentation in color images of the retina." IEEE Transactions on Medical Imaging, 23(4), 501-509.

[27] Ting, D. S., et al. (2017). "Artificial intelligence and deep learning in ophthalmology." British Journal of Ophthalmology, 103(2), 167-175.

[28] Xu, Y., et al. (2016). "Retinal fundus image registration via vascular structure graph matching." IEEE Transactions on Biomedical Engineering, 63(12), 2598-2609.

[29] Akram MU, Abdul Salam A, Khawaja SG, Naqvi SGH, Khan SA. RIDB: A Dataset of fundus images for retina-based person identification. Data Brief. 2020 Oct 20;33:106433. doi: 10.1016/j.dib.2020.106433. PMID: 33209967; PMCID: PMC7658644.

[30]Chen, B. C., et al. (2014). "Expanding the dynamic range of fluorescent reporters by brightening." Nature Biotechnology, 32(3), 264-267.

[31]Ding, Y., et al. (2015). "Deep learning for tissue microarray image-based outcome prediction in patients with colorectal cancer." Scientific Reports, 5(1), 1-11.

[32]Prodan, A., et al. (2020). "Microbial players in the gut-lung axis and their role in respiratory tract infections." Virulence, 11(8), 965-981.

[33]Colling, R., et al. (2018). "Artificial intelligence in digital pathology: A roadmap to routine use in clinical practice." Journal of Pathology Informatics, 9(1), 1-7.

[34]Ehteshami Bejnordi, B., et al. (2017). "Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer." JAMA, 318(22), 2199-2210.

[35]Kononenko, I. (2019). "Machine Learning for Medical Diagnosis: History, State of the Art and Perspective." Artificial Intelligence in Medicine, 98, 3-18.

[36]Roth, R., et al. (2014). "Pypotrace: A Potrace Binding for Python." SoftwareX, 1, 19-25.

[37] Wu S, Okada R, Liu Y, Fang Y, Yan F, Wang C, Li H, Kobayashi H, Chen Y, Tang Q. Quantitative analysis of vascular changes during photoimmunotherapy using speckle variance optical coherence tomography (SV-OCT). Biomed Opt Express. 2021 Mar 2;12(4):1804-1820. doi: 10.1364/BOE.419163. PMID: 33996199; PMCID: PMC8086455.

[38]Taieb, A., & Picardo, M. (2009). "The definition and assessment of vitiligo: A consensus report of the Vitiligo European Task Force." Pigment Cell & Melanoma Research, 22(5), 567-571.

[39]Ezzedine, K., et al. (2015). "Vitiligo." The Lancet, 386(9988), 74-84.

[40]Ronneberger, O., Fischer, P., & Brox, T. (2015). UNet: Convolutional Networks for Biomedical Image Segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI) (pp. 234-241). Springer.

[41]Zheng, Y., Liu, H., Hu, Y., Li, C., & Zhao, Z. (2020). A Review of Semantic Segmentation Using Deep Learning in Medical Imaging. Journal of Medical Imaging and Health Informatics, 10(7), 1607-1616.

[42]Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A., Ciompi, F., Ghafoorian, M., ... & Sanchez, C. I. (2017). A survey on deep learning in medical image analysis. Medical Image Analysis, 42, 60-88.

[43]Liskowski, P., & Krawiec, K. (2016). Segmenting retinal blood vessels with deep neural networks. IEEE Transactions on Medical Imaging, 35(11), 2369-2380.

[44]Antropova, N., Huynh, B. Q., Giger, M. L., & Horsch, K. (2017). Deep convolutional neural networks for breast cancer histology image analysis. In Medical Imaging 2017: Digital Pathology (Vol. 10140, p. 101400U). International Society for Optics and Photonics.

[45]Zahiri, M., Diamant, I., Gavves, E., & Gevers, T. (2018). Deep learning for automatic localization, identification, and segmentation of vertebral bodies in volumetric CT scans. Medical Image Analysis, 44, 236-249.

[46]Thakur, K., Thakur, P., Kumar, S., & Dhiman, A. (2021). A review of deep learning-based segmentation techniques for medical images. Journal of Ambient Intelligence and Humanized Computing, 1-27.

[47]Ronneberger, O., & Welling, M. (2015). UNet++: A Nested UNet Architecture for Medical Image Segmentation. In Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support (pp. 3-11). Springer.

[48]Hillmer, D., Merhi, R., Boniface, K., Barnetche, T., Seneschal, J., & Hagedorn, M. (2023). Evaluation of facial vitiligo severity with a mixed clinical and artificial intelligence approach. Journal of Investigative Dermatology, 143(8), 1723-1725. https://doi.org/10.1016/j.jid.2023.07.014

[50]Reference: Swain, M. J., & Ballard, D. H. (1991). Indexing via color histograms. International Journal of Computer Vision, 7(1), 11-32.

[51]E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, “Photographic tone reproduction for digital images,” ACM Transactions on Graphics, vol. 21, no. 3, pp. 267–276, 2002.

[52]E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, “Color transfer between images,” IEEE Computer Graphics & Applications, vol. 21, no. 5, pp. 34–41, 2001.

[53]Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273-297.

[54]: Schölkopf, B., & Smola, A. J. (2002). Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press.

[55]: Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-32.

[56]: Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R News, 2(3), 18-22.

[57]Liao, P-S., Chen, T-S. and Chung, P-C., “A fast algorithm for multilevel thresholding”, Journal of Information Science and Engineering 17 (5): 713-727, 2001. Available at: <https://ftp.iis.sinica.edu.tw/JISE/2001/200109_01.pdf>

[58] Cannatella, David & Sá, Rafael. (1993). Xenopus Laevis as a Model Organism. Systematic Biology - SYST BIOL. 42. 10.2307/2992485.

[59] Akram MU, Abdul Salam A, Khawaja SG, Naqvi SGH, Khan SA. RIDB: A Dataset of fundus images for retina-based person identification. Data Brief. 2020 Oct 20;33:106433. doi: 10.1016/j.dib.2020.106433. PMID: 33209967; PMCID: PMC7658644.

[60] Dice, L. R. (1945). Measures of the Amount of Ecologic Association Between Species. Ecology, 26(3), 297-302. Link

[61] Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 7132-7141). Link

[62] Ismail Fawaz, H., Lucas, B., Forestier, G., Pelletier, C., & Schmidt, D. (2019). Attention gated networks: Learning to leverage salient regions in medical images. Medical Image Analysis, 53, 197-207. Link

[63] Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L., & Muller, P. A. (2018). Deep learning for time series classification: a review. Data Mining and Knowledge Discovery, 32(4), 1-47. Link

[64] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639), 115-118. Link

[65] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), Vol. 1, pp. I-511 - I-518, 2001.

[66] Tanno, Y. (2004). Adaptive Line Integral Convolution for Visualizing Diffusion Tensor Images. Retrieved from http://www.image.med.osaka-u.ac.jp/member/yoshi/paper/linefilter.pdf

Appendix

ORIGINAL ARTICLE

Evaluation of Facial Vitiligo Severity with a Mixed Clinical and Artificial Intelligence Approach

Dirk Hillmer[1], Ribal Merhi[2], Katia Boniface[2], Alain Taieb[1], Thomas Barnetche[3], Julien Seneschal[2,4,5] and Martin Hagedorn[1,5]

Vitiligo is the most common depigmenting skin disorder. Given the ongoing development of new targeted therapies, it has become important to evaluate adequately the surface area involved. Assessment of vitiligo scores can be time consuming, with variations between investigators. Therefore, the aim of this study was to build an artificial intelligence system capable of assessing facial vitiligo severity. One hundred pictures of faces of patients with vitiligo were used to train and validate the artificial intelligence model. Sixty-nine additional pictures of facial vitiligo were then used as a final dataset. Three expert physicians scored the facial vitiligo on the same 69 pictures. Inter and intrarater performances were evaluated by comparing the scores between raters and artificial intelligence. Algorithm assessment achieved an accuracy of 93%. Overall, the scores reached a good agreement between vitiligo raters and the artificial intelligence model. Results demonstrate the potential of the model. It provides an objective evaluation of facial vitiligo and could become a complementary/alternative tool to human assessment in clinical practice and/or clinical research.

Journal of Investigative Dermatology (2024) **144,** 351–357; doi:10.1016/j.jid.2023.07.014

INTRODUCTION

Vitiligo is the most common acquired skin depigmenting disorder, and it considerably impairs QOL (Boniface et al., 2018; Ezzedine et al., 2015). Management of the disease remains challenging, with only limited therapeutic options available. However, recent mechanistic insights have led to the development of new treatments. For example, ruxolitinib cream, a topical Jak inhibitor, is now approved for the disease (Rosmarin et al., 2022). In clinical practice and/or clinical research, vitiligo is often evaluated using validated scores (e.g., Vitiligo Area Scoring Index and the Vitiligo Extent Score) (Hamzavi et al., 2004; van Geel et al., 2016). These scores reliably assess depigmentation severity and extent as well as response to treatment. Recently, a modification of the Vitiligo Area Scoring Index score was validated for the assessment of facial involvement (Bae et al., 2022). The Face Vitiligo Area Scoring Index score is now used as a primary outcome measure in clinical trials that evaluate treatment efficacy (Rosmarin et al., 2022). However, despite their value in assessing treatment response, both scores are time consuming.

With regard to digital assessment of vitiligo lesions, a recent systematic review showed that manual tracing on transparent sheets combined with digital measurement could be considered the best validated method to assess the measurement of target lesions (van Geel et al., 2022). However, no robust digital image analysis system has been validated to assess a global evaluation of specific areas. Therefore, an objective, computer-based automatic scoring method would be easier to use, would be time saving, and would allow a more precise global evaluation than human analysis. A machine learning algorithm previously achieved results similar to those of dermatologists for the detection of skin cancer (Esteva et al., 2017). In addition, a machine learning algorithm can also be used to assess objective scores in chronic inflammatory skin diseases (e.g., for psoriasis) (Meienberger et al., 2020). However, despite this progress, research on neural network–based machine learning algorithms to assess vitiligo is lacking.

The objective of this study was to build and evaluate a neural network allowing the detection of vitiligo lesions on photographs with a special interest in lesions located on the face.

RESULTS

Assessing the severity of facial vitiligo with artificial intelligence

The artificial intelligence (AI) algorithm was designed as described in the Materials and Methods. First, it extracted the face from a clinical picture and created a vitiligo mask

[1]*BRIC (BoRdeaux Institute of onCology), INSERM UMR1312, Team 5, University of Bordeaux, Bordeaux, France;* [2]*CNRS, UMR 5164, Immuno ConcEpT, University of Bordeaux, Bordeaux, France;* [3]*Department of Rheumatology, National Reference Center for Rare Systemic Autoimmune Diseases, FHU ACRONIM, Pellegrin Hospital, CHU de Bordeaux, Bordeaux, France; and* [4]*Department of Dermatology and Pediatric Dermatology, National Reference Center for Rare Skin Disorders, UMR 5164, Saint-André Hospital, University Hospital of Bordeaux, Bordeaux, France*

[5]*These authors contributed equally to this work.*

Correspondence: Julien Seneschal, Department of Dermatology and Pediatric Dermatology, National Reference Center for Rare Skin disorders, UMR 5164, Hospital Saint-André, Bordeaux University, Bordeaux F-33000, France. E-mail: julien.seneschal@chu-bordeaux.fr

Abbreviations: AI, artificial intelligence; CI, confidence interval; CNN, convolutional neural network; HSV, hue, saturation, value/luminance

Received 17 February 2023; revised 18 July 2023; accepted 27 July 2023; accepted manuscript published online 15 August 2023; corrected proof published online 26 September 2023

(Figure 1). Next, it generated the convolutional neural network (CNN) training and testing data. It was trained and tested using 100 vitiligo face images that were divided into a training set (80%) and a validation set (20%). After training, the CNN reached an accuracy of 93%.

As shown in Figure 1c, the resulting picture was a composition of three images starting from left to right with the original image, the calculated vitiligo mask, and an overlay image of the original image plus the vitiligo mask image. The pictures of vitiligo mask contained a calculated percentage corresponding to the estimated depigmented area of the face ranging from 0% (no depigmentation) to 100% (complete depigmentation) (Figure 2a). The evolution of this percentage over time could be used in clinical practice to identify the positive or negative impact of treatments during patient follow-up (Figure 2b).

Comparison between AI and physician assessment of facial vitiligo

Sixty-nine images of facial vitiligo were selected as a final dataset. The images corresponded to 59 patients with facial vitiligo, of whom 10 were photographed twice at 6-month intervals. Table 1 shows the demographic and clinical characteristics of the included 59 patients. All patients were mainly followed for nonsegmental vitiligo (6.8% acrofacial, 91.5% generalized), except one patient who presented with facial segmental vitiligo.

Three expert physicians assessed the vitiligo-affected facial surface area at two time points. The surface of the vitiligo area was scored from 0% (no vitiligo) to 100% (complete depigmentation of the area). As shown in Table 2, the overall inter-rater and intrarater reliability was excellent with interclass correlation coefficients of 0.80 (95% confidence interval [CI] = 0.71−0.80) and 0.81 (95% CI = 0.78−0.90), respectively. The assessment of facial vitiligo reached a good agreement between rater 1 and AI (0.70, 95% CI = 0.52−0.82), rater 2 and AI (0.67, 95% CI = 0.48−0.80), and rater 3 and AI (0.69, 95% CI = 0.51−0.81). Correlation analysis between the mean score of all physicians and the AI showed good data clustering around the regression line of the scatter plot (Supplementary Figure S1). In addition, the evolution of the vitiligo lesions analyzed by the raters for the 10 patients who got two sets of pictures significantly correlated with the evolution identified by the AI (rater 1 vs. AI: $r = 0.84$, $P = 0.002$; rater 2 vs. AI: $r = 0.86$, $P = 0.001$; rater 3 vs. AI: $r = 0.90$, $P = 4 \times 10^{-4}$). However, AI values were not always directly comparable with the expert physicians' assessment, especially for patients with limited lesions or in contrast for patients with large surfaces involved. These differences could be due to image acquisition or a tendency for human deviation based on experience and the global estimation of the area in question (Supplementary Figure S2a and b).

DISCUSSION

As a proof of concept study, our model was first trained and validated to assess facial involvement of vitiligo. Facial vitiligo has a major impact on QOL of patients. Indeed, previous research showed that vitiligo on visible areas such as the face represents an extreme burden for patients (Narayan et al., 2021). The face is also one of the most commonly affected areas by vitiligo (Ezzedine et al., 2015). With regard to patients' perspective, successful repigmentation of vitiligo, especially on the face, should be >80% repigmentation, as recently reported (Eleftheriadou et al., 2019). Therefore, because lesions located on the face respond usually better than those on other areas (Whitton et al., 2015), it remains important to adequately assess facial vitiligo lesions and the response to treatments justifying the development of our model first for the evaluation of the face. In addition, the most recent clinical trials evaluating topical Jak inhibitors (e.g., ruxolitinib) (Rosmarin et al., 2022, 2020) or systemic Jak inhibitors (Ezzedine et al., 2023) for vitiligo assessed the efficacy of the therapy for the improvement of facial lesions as a primary outcome, confirming the need of a reliable and objective method to evaluate vitiligo lesions on the face. Once our model is validated for evaluation of facial involvement in vitiligo, the next step will be to train it to analyze full-body images of patients with vitiligo.

Clinical evaluation of vitiligo severity according to the affected body surface area remains subjective and may vary between physicians. Such subjectivity can be circumvented using machine learning supported by faster computers and the capacity for large networks to make rapid calculations. For instance, the global evaluation of facial vitiligo by our physicians may have led to overestimation, especially in patients with large surface areas involved. Because this problem would not have affected the AI model, some of the differences in assessment between AI and the raters could be explained.

Although the use of machine learning in dermatology has improved in recent years, most studies have sought to evaluate pigmented skin lesions and melanoma (Esteva et al., 2017; Lyakhov et al., 2022; Shetty et al., 2022). Data are sparse for depigmenting lesions such as vitiligo. After training, our CNN achieved an accuracy of 93%. Two recent studies reported deep learning models that achieved a sensitivity of 90 and 92% for classification of vitiligo (Guo et al., 2022; Yasir et al., 2014).

Identifying novel concepts for image analysis of vitiligo lesions thus remains a major challenge in AI, which usually functions well in limited settings and for specific tasks. Although pretrained models are already available for facial detection, they are not suitable for detecting vitiligo. Furthermore, applying such AI models for the facial detection of patients with vitiligo could potentially lead to false recognition owing to the presence of depigmented areas. To date, development of neural networks for vitiligo has been mainly done for the detection of vitiligo lesions but not for a global evaluation of a specific area. For instance, Khatibi et al. (2021) developed an accurate method for this task, which was supported by a complex combination of different deep neural networks and conventional models for the detection of vitiligo. In addition, the amount of training data and the quality of the images are critical for constructing reliable neural networks. AI models need to be continuously enriched with additional data obtained from clinical practice, improving their quality and accuracy. In multicultural populations with different skin phototypes, skin tone differs from one person to another, so the system would have to be

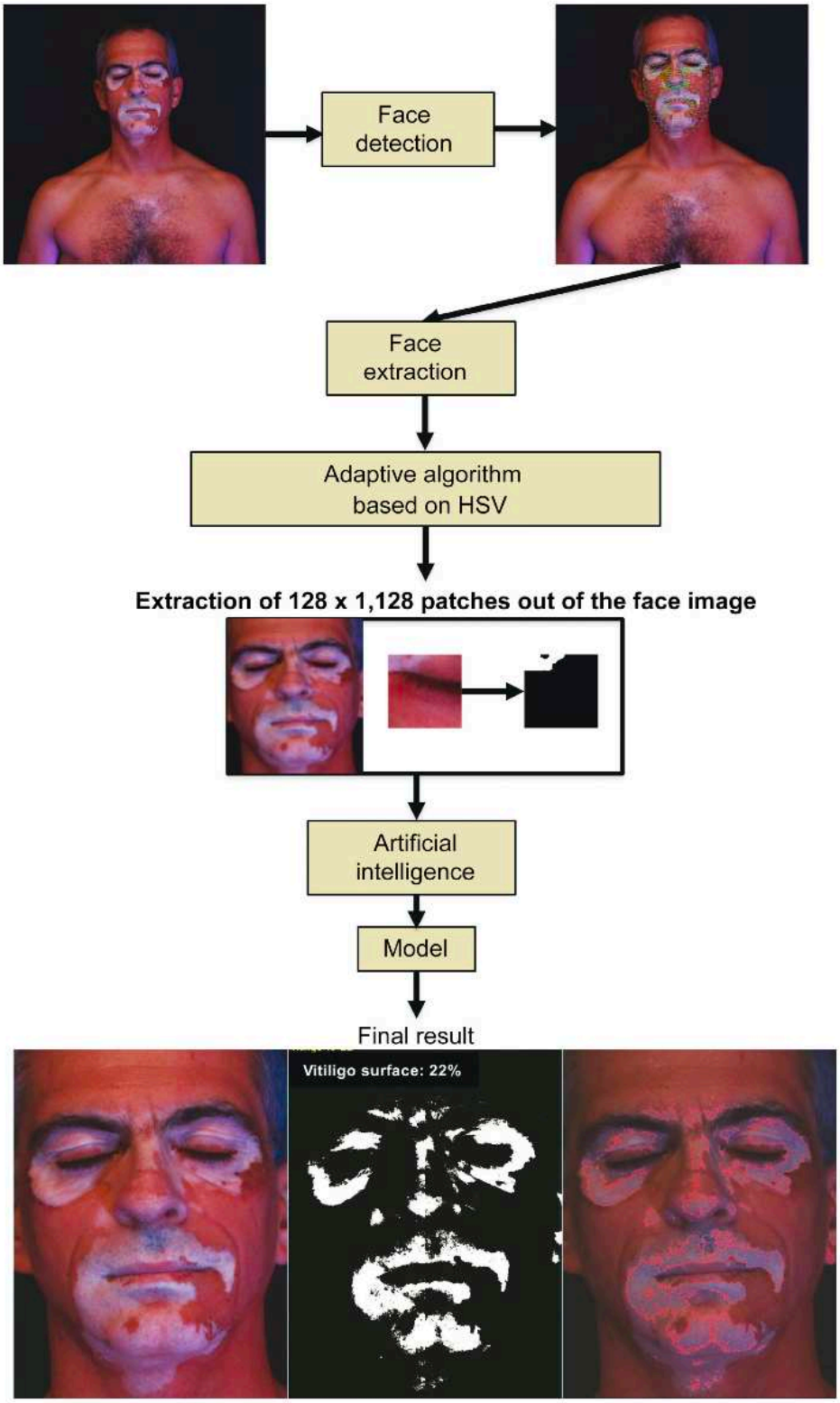


Figure 1. Image processing of facial vitiligo. Original clinical pictures of the face were taken with FOTOFINDER bodyscan system combined with two high-output UV flashes. Vitiligo mask was created by algorithm. The algorithm extracts face from clinical pictures and creates a vitiligo mask as well as creates CNN training and testing data. The final result shows the original image with face location landmarks (left), the final processed result (center), and superimposition of vitiligo edges on the original image (right). CNN, convolutional neural network. Images published with patient consent.

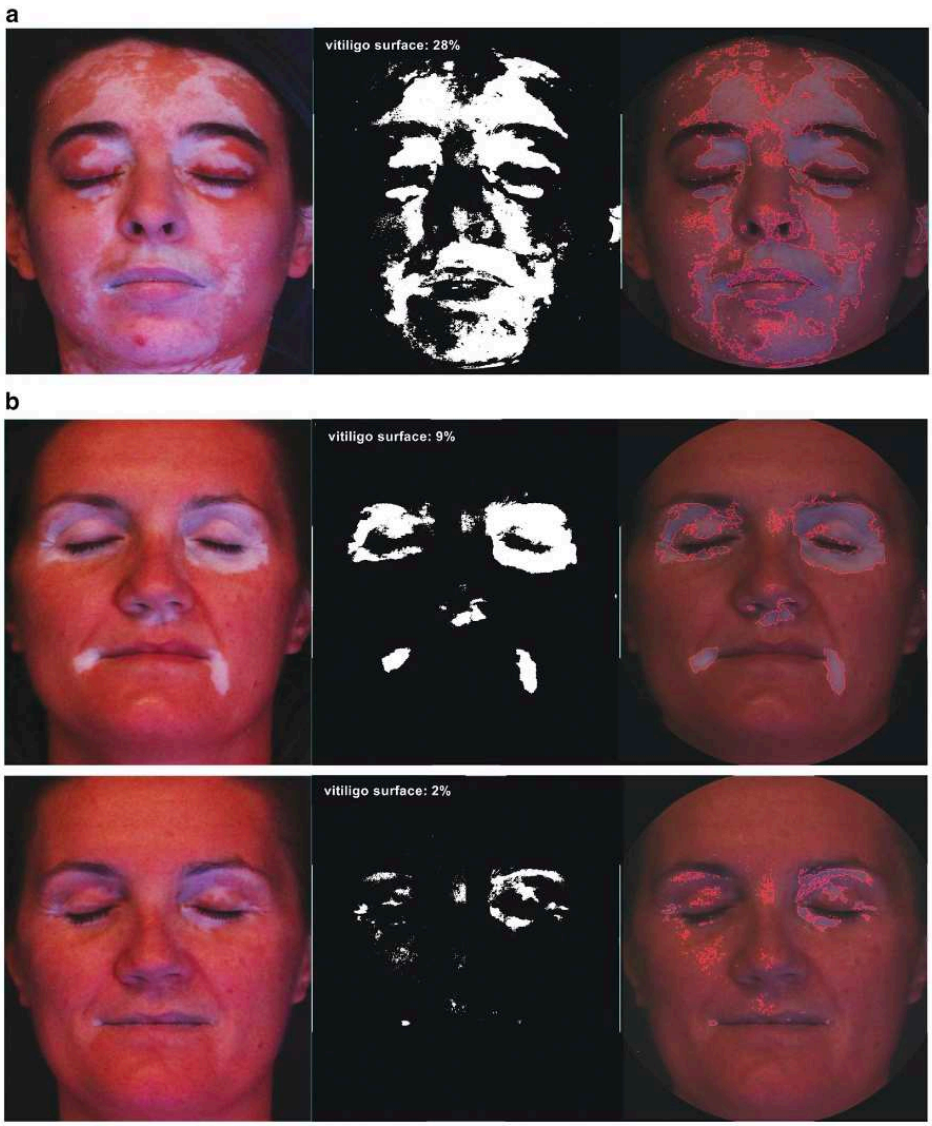


Figure 2. Evaluation of facial vitiligo surface (**a**) Panel shows the surface of facial vitiligo estimated by algorithm. (**b**) Evaluation of surface area involved at two time points in the same patient. Images published with patient consent.

flexible and adjustable, especially for the detection of vitiligo. The reuse of already trained models and the possibility to load them in other U-net models could help to improve vitiligo detection. Again, however, differences in skin color between populations would pose a challenge for this procedure. For example, Guo et al. (2022) developed a deep-learning–based approach that assessed the severity of vitiligo-target lesions in an Asian population. They took clinical photographs of patients with phototypes type III or IV using a digital camera without UV flash or blue filtering. However, despite the interest in the procedure, their training data and the models they created cannot be used in Caucasian people because their skin tones are quite different from those of Asian subjects. Similarly, the model developed by Khatibi et al. (2021) was validated on pictures taken with normal light from a limited number of patients sharing similar skin tones, limiting the training of the method.

A strength of our study is that the clinical pictures of vitiligo were taken with a standardized position during routine visits. The clinical procedure is thus more reliable, and the results are rapidly comparable. For this reason, the method could be used to analyze the degree of depigmentation during visits and to follow the evolution of vitiligo after treatment, thereby helping physicians in everyday practice. In addition, the method could be used in clinical research trials for objective and reproducible evaluation of facial vitiligo before and after therapeutic interventions.

The main limitation of this study concerns the quality of the images obtained during clinical practice in a short period of time. Although a standardized face-on position was adopted by all patients, the distance between the patient and the camera as well as the picture angle were not exactly the same. To address this limitation, the patient could be positioned in a pre-established position, although this would be

Table 1. Demographic and Clinical Characteristics of Patients with Vitiligo

Characteristics	Vitiligo (n = 59)
Sex, n (%)	
Female	41 (69.5)
Male	18 (30.5)
Age at inclusion, y	
Mean (±SD)	38.7 (±14.05)
Range	21–65
Skin phototype	
I, II, III	45 (76.3)
IV, V, VI	14 (23.7)
Age at vitiligo onset, y	
Mean (±SD)	24.20 (±14.53)
Range	6–56
Disease duration, y	
Mean (±SD)	14.20 (±10.08)
Type of vitiligo, n (%)	
Acrofacial	4 (6.8)
Generalized	54 (91.5)
Segmental	1 (1.7)
Personal history of autoimmune or autoinflammatory disease, n (%)	
Autoimmune thyroiditis	14 (23.7)
Other chronic inflammatory diseases (including atopic disease, psoriasis, type 1 diabetes, and alopecia areata)	13 (22.03)
Affected BSA, %, mean (±SD)	9.95 (±9.61)
Vitiligo Extent Score mean (±SD)	9.12 (±9.82)
Disease activity, n (%)	
Regressive	4 (6.9)
Stable	24 (40.6)
Active	31 (52.5)

Abbreviations: BSA, body surface area; y, year.

time consuming in clinical practice. Another limitation is that the image quality and therefore image processing were impacted by the UV flash used to enhance the contrast between normal and depigmented skin. This flash creates a shadowy area, which affects pixel values and the evaluation of depigmented areas. In addition, some skin areas may be overexposed and appear white, leading to confusion with vitiligo lesions. However, these technical issues can be addressed if an adjusted dynamic boundary based on the image data is applied. The hue, value, saturation channels can then be set almost optimally. Nevertheless, this limitation may account for some of the differences observed between AI and the raters in patients with a small involved surface area because there may have been false-positive detection of vitiligo lesions. In conclusion, this validated AI model to assess facial vitiligo could become a useful tool in clinical practice and clinical research trials.

Table 2. Assessment of Inter and Intrarater Reliability

Inter-Rater Reliability	ICC (95% CI)	Intrarater Reliability	ICC (95% CI)
Overall raters (n = 3)	0.80 (0.71–0.87)	Overall raters (n = 3)	0.81 (0.78–0.90)
Rater 1 versus AI	0.70 (0.52–0.82)	Rater 1	0.96 (0.94–0.98)
Rater 2 versus AI	0.67 (0.48–0.80)	Rater 2	0.96 (0.93–0.98)
Rater 3 versus AI	0.69 (0.51–0.81)	Rater 3	0.98 (0.97–0.99)

Abbreviations: AI, artificial intelligence; CI, confidence interval; ICC, interclass correlation coefficient.

MATERIALS AND METHODS

Patients

This retrospective, noninterventional, single-center study was performed in the Department of Dermatology of the University Hospital of Bordeaux (Bordeaux, France) and was approved by the local ethics committee of the University of Bordeaux. All patients received a patient information sheet and gave oral consent. Patients who accepted to be published provided written consent. Clinical pictures of the face were taken with the FOTOFINDER bodyscan system combined with two high-output UV flashes. The photographs were standardized in a neutral position.

One hundred facial vitiligo pictures were selected, of which 80% were used for training, and the remaining 20% was used for validation of the trained model. As a final test dataset, 69 additional new facial pictures were used. They were taken from 59 adult patients, of whom 10 were re-evaluated 6 months after an intervention therapy. Three expert physicians estimated the proportion of vitiligo lesions on the face at two time points. Vitiligo was assessed as 0% (no vitiligo lesion) to 100% (complete depigmentation). Inter and intrarater reliability were evaluated by comparing the scores between experts at each evaluation and between two different time points (test–retest), respectively. A two-way random model was performed for inter-raters, and a two-way mixed model was performed for intraraters. Absolute-agreement and single-measure intraclass correlation coefficients were also estimated. Inter-rater reliability was evaluated between experts and AI. Logarithmic transformation was performed for non-normally distributed data. The following guidelines for the interpretation of the interclass correlation coefficients were used: <0.4 was considered poor, 0.4–0.59 was considered fair, 0.6–0.74 was considered good, and ≥0.75 was considered excellent (Cicchetti, 1994).

Data preparation

Neuronal network architecture. The AI algorithm extracted the face from the clinical picture and then created a vitiligo mask. Next, it generated the CNN training and testing data using the Tensorflow Keras library. Patches of the images were created to generate training data for the CNN. After testing the performance of the network with different patch sizes, a 128 × 128 size was chosen. In total, 15,000 image patches were created from the clinical pictures used for the training set (80% of the 100 facial vitiligo images).

Data generation, evaluation, and data analysis

Delta analysis. The image containing the patient's head was used for processing. Because pictures taken with UV flashes produced some shadows on the pictures, the image was processed in two patches: one from the left and the other from the right. The mean of the patches was used to estimate the difference in lighting.

Locating background. Background location was necessary to separate pixels that were connected to the patient. A color model is an abstract model for describing the representation of colors as tuples of numbers. It usually consists of three or four values known as color components or color coordinates.

Hue, saturation, value/luminance (HSV) (Weatherall and Coombs, 1992) is a model that presents similar colors grouped closely together. It is often used in graphic applications and for image analysis. HSV stands for hue (color/hue), saturation (saturation), and value/luminance (lightness/luminosity, not to be confused with brightness) (Supplementary Figure S3). Hue sets the position of the color on the color wheel (from 0′ to 360′), saturation is the percentage of saturation (0–100%), and value is the percentage of luminosity (0–100%). To detect all or most skin tones, a set of images was processed, and the upper and lower boundaries were adjusted. To collect human skin with as few false positives as possible, we set the HSV model as described below. Owing to hair color and other confounding factors, an accuracy of 100% cannot be achieved.

```
# Defining HSV Thresholds
lower_threshold [0, 48, 80]
upper_threshold [20, 255, 255]
```

Background subtraction created a mask for the pixels that fell within the boundaries. By removing the other pixels and setting them to black, a color image of the remaining skin can be obtained.

Locating face. The image produced in the previous step was analyzed with Mediapipe Python (https://google.github.io/mediapipe/getting_started/python.html) (Figure 1). It allowed us to detect faces and establish facial landmarks. Alternatively, one could use dlib (https://pypi.org/project/dlib/), which is a library for image recognition but is less accurate than Mediapipe. Using landmark detection, that is, the location of essential points of interest, we were able to locate typical features of the image such as the eyes and nose. Figure 1a and b shows the original image and the location of the face, respectively.

The information was then used to modify the image if it was not centered. The position of the nose and the left and right sides of the skin borders were used to split the image along the nose line so that both parts were of equal size. This allowed the difference in exposure during processing to be taken into account. Figure 1c shows the result: a combination of the original facial image concatenated with the vitiligo mask, revealing the superposition of the edges of the vitiligo on the original image.

Head size. We then used a mask containing information about the position of the face and placed an ellipse around the remaining pixels. The total head size to be used to assess the extent of the vitiligo could then be calculated.

Calculating the extent of vitiligo. In the event of lighting differences, the image was split, and each side was processed with different parameters, as shown below.

```
# Defining HSV vitiligo range
lower_threshold.[40, s_min, 61]
upper_threshold. [178, s_max, 255]
```

The correct skin tone was defined before applying the AI. Owing to the UV lighting, the skin had a specific color, and the nonpigmented areas were more exposed. On the basis of the HSV color schema and especially the saturation channel, the lower S channel marked the start of the area of vitiligo, and the upper S channel ended it. Therefore, we used an iterative approach by changing the lower S channel value (s_min) until the pixels in the image were no longer visible. This step provided information about skin tone. The algorithm created a mask that it then superimposed onto the face and vitiligo lesions. This way, we were able to calculate the percentage of white pixels classified as vitiligo on the basis of skin color. The mask generated was used to create a dataset for machine learning.

Machine learning. A supervised learning algorithm was used, where the supervision was performed by automatic processing. This was achieved by taking 128 × 128 patches out of the image and the mask. Next, a sequential CNN and a U-Net were developed. The CNN had an accuracy of 93%. Instead of using the ReLu (Rectified Linear Unit) function, we used a sigmoid function, that is, a specific form of a logistic function.

Using the U-net extended the training time to about 5 minutes per epoch, and an accuracy of 76% was reached using black and white patches. The detailed performance of the networks can be found in Supplementary Materials and Methods. The approach was as follows: first, an image was separated into patches, and each patch was then classified by the U-net as either pigmented skin or vitiligo. If the U-net prediction was over 70%, the patch was counted as vitiligo. Otherwise, the patch was considered normal pigmented skin. This reduced the number of vitiligo pixels in the result. Therefore, the CNN provided help with the decision regarding whether the area was more likely to be normal pigmented skin or vitiligo. The model thus demonstrated its high level of confidence in differentiating normal and vitiligo skin.

Data availability statement

The authors confirm that the data supporting the findings of this study are available in the article and its supplementary materials, including the code of the model. Datasets related to this article are available upon request (martin.hagedorn@u-bordeaux.fr or julien.seneschal@chu-bordeaux.fr) and are hosted by the University Hospital of Bordeaux. The data are not publicly available because they contain information that could compromise the privacy of research participants.

ORCID

Julien Seneschal: http://orcid.org/0000-0003-1139-0908

CONFLICT OF INTEREST

The authors state no conflict of interest.

ACKNOWLEDGMENTS

The auhtors thank Ray Cooke of the University of Bordeaux for copyediting.

AUTHOR CONTRIBUTIONS

Conceptualization: DH, AT, JS, MH; Data Curation: DH, JS, MH; Formal Analysis: DH, TB, JS, MH; Investigation: DH, RM, KB, AT, JS, MH; Methodology: DH, JS, MH; Project Administration: JS, MH; Resources: JS, MH; Software: DH; Supervision: JS, MH; Validation: JS, MH; Visualization: JS, MH; Writing - Original Draft Preparation: DH, JS, MH; Writing - Review and Editing: CM, LM, AT, TB, JS, MH

SUPPLEMENTARY MATERIAL

Supplementary material is linked to the online version of the paper at www.jidonline.org, and at https://doi.org/10.1016/j.jid.2023.07.014.

REFERENCES

Bae JM, Zubair R, Ju HJ, Kohli I, Lee HN, Eun SH, et al. Development and validation of the fingertip unit for assessing facial vitiligo area scoring index. J Am Acad Dermatol 2022;86:387–93.

Boniface K, Seneschal J, Picardo M, Taïeb A. Vitiligo: focus on clinical aspects, immunopathogenesis, and therapy. Clin Rev Allergy Immunol 2018;54:52–67.

Cicchetti DV. Guidelines, criteria, and rules of thumb for evaluating normed and standardized assessment instruments in psychology. Psychol Assess 1994;6:284–90.

Eleftheriadou V, Hamzavi I, Pandya AG, Grimes P, Harris JE, Huggins RH, et al. International Initiative for Outcomes (INFO) for vitiligo: workshops with patients with vitiligo on repigmentation. Br J Dermatol 2019;180:574–9.

Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM, Blau HM, et al. Dermatologist-level classification of skin cancer with deep neural networks [published correction appears in Nature 2017;546:686]. Nature 2017;542:115–8.

Ezzedine K, Eleftheriadou V, Whitton M, van Geel N. Vitiligo. Lancet 2015;386:74–84.

Ezzedine K, Peeva E, Yamaguchi Y, Cox LA, Banerjee A, Han G, et al. Efficacy and safety of oral ritlecitinib for the treatment of active nonsegmental vitiligo: a randomized phase 2b clinical trial. J Am Acad Dermatol 2023;88:395–403.

Guo L, Yang Y, Ding H, Zheng H, Yang H, Xie J, et al. A deep learning-based hybrid artificial intelligence model for the detection and severity assessment of vitiligo lesions. Ann Transl Med 2022;10:590.

Hamzavi I, Jain H, McLean D, Shapiro J, Zeng H, Lui H. Parametric modeling of narrowband UV-B phototherapy for vitiligo using a novel quantitative tool: the Vitiligo Area Scoring Index. Arch Dermatol 2004;140:677–83.

Khatibi T, Rezaei N, Ataei Fashtami LA, Totonchi M. Proposing a novel unsupervised stack ensemble of deep and conventional image segmentation (SEDCIS) method for localizing vitiligo lesions in skin images. Skin Res Technol 2021;27:126–37.

Lyakhov PA, Lyakhova UA, Nagornov NN. System for the recognizing of pigmented skin lesions with fusion and analysis of heterogeneous data based on a multimodal neural network. Cancers 2022;14:1819.

Meienberger N, Anzengruber F, Amruthalingam L, Christen R, Koller T, Maul JT, et al. Observer-independent assessment of psoriasis-affected area using machine learning. J Eur Acad Dermatol Venereol 2020;34:1362–8.

Narayan VS, Uitentuis SE, Luiten RM, Bekkenk MW, Wolkerstorfer A. Patients' perspective on current treatments and demand for novel treatments in vitiligo. J Eur Acad Dermatol Venereol 2021;35:744–8.

Rosmarin D, Pandya AG, Lebwohl M, Grimes P, Hamzavi I, Gottlieb AB, et al. Ruxolitinib cream for treatment of vitiligo: a randomised, controlled, phase 2 trial. Lancet 2020;396:110–20.

Rosmarin D, Passeron T, Pandya AG, Grimes P, Harris JE, Desai SR, et al. Two Phase 3, randomized, controlled trials of Ruxolitinib cream for vitiligo. N Engl J Med 2022;387:1445–55.

Shetty B, Fernandes R, Rodrigues AP, Chengoden R, Bhattacharya S, Lakshmanna K. Skin lesion classification of dermoscopic images using machine learning and convolutional neural network [published correction appears in Sci Rep 2022;12:21919]. Sci Rep 2022;12:18134.

van Geel N, Lommerts J, Bekkenk M, Wolkerstorfer A, Prinsen CAC, Eleftheriadou V, et al. Development and validation of the Vitiligo Extent Score (VES): an international collaborative initiative. J Invest Dermatol 2016;136:978–84.

Van Geel N, Saeys I, Causenbroeck JV, Duponselle J, Grine L, Pauwels N, et al. Image analysis systems to calculate the surface area of vitiligo lesions: a systematic review of measurement properties. Pigm Cell Melanoma R 2022;35:480–94.

Weatherall IL, Coombs BD. Skin color measurements in terms of CIELAB color space values. J Invest Dermatol 1992;99:468–73.

Whitton ME, Pinart M, Batchelor J, Leonardi-Bee J, González U, Jiyad Z, et al. Interventions for vitiligo. Cochrane Database Syst Rev 2015;2:CD003263.

Yasir R, Rahman MdA, Ahmed N. Dermatological disease detection using image processing and artificial neural network. Paper presented at: 8th International Conference on Electrical and Computer Engineering. 2014; Dhaka, Bangladesh.

SUPPLEMENTARY MATERIALS AND METHODS

Convolutional neural network training

Creation of training data. The image is processed using a converted hue, saturation, value/luminance image. The image channels are then checked separately, and the high and low boundaries are set. The algorithm is as follows:

Locate the skin in the image to remove the background. Not all backgrounds can be removed, so there will be some noise left.

Locate the face in the image.

Locate the minimum V value as min_v.

Locate the minimum S value as min_s. Assume the maximum S value to be 42 values higher.

Locate the maximum H value as max_h

Create an output mask with the following settings:

Low H, S, V Values as (0, min_s, min_v)

Hight H, S, V Values as (max_h,max_s, 252)

The values for min_h and max_v can be set to default values because they have little impact on the color selection.

The generated mask and the extracted head image or normal image can now be processed to generate convolutional neural network training and test data.

The information on the mask is used to decide whether the image patch can be classified as vitiligo or as normal pigmented skin. The patch size is currently set to 128 × 180 but can be extended.

Image patches are then saved in different orientations.

The information is then stored in separate folders.

The convolutional neural network test and training data use the folders to classify the skin.

The training of the network and validation of the trained network are defined when loading the data.

Convolutional neural network training and validation. Artificial intelligence model used and training results based on created vitiligo dataset are shown as follows:

Model: "sequential"

Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 80, 80, 3)	0
conv2d (Conv2D)	(None, 78, 78, 32)	896
max_pooling2d (MaxPooling2D	(None, 39, 39, 32)	0)
conv2d_1 (Conv2D)	(None, 37, 37, 32)	9248
max_pooling2d_1 (MaxPooling	(None, 18, 18, 32)	0 2D)
conv2d_2 (Conv2D)	(None, 16, 16, 32)	9248
max_pooling2d_2 (MaxPooling	(None, 8, 8, 32)	0 2D)
flatten (Flatten)	(None, 2048)	0
dense (Dense)	(None, 128)	262272
dense_1 (Dense)	(None, 2)	258
Total params: 281,922		
Trainable params: 281,922		
Non-trainable params: 0		
Train on 250 steps, validate on 63 steps		
Epoch 1/30		
249/250 [==============================>.] - ETA: 0s - batch: 124.0000 - size: 1.0000 - loss: 0.3801 - acc: 0.8389		
250/250 [==============================] - 7s 26ms/step - batch: 124.5000 - size: 1.0000 - loss: 0.3824 - acc: 0.8387 - val_loss: 0.3129 - val_acc: 0.8796		
Epoch 30/30		
250/250 [==============================] - 6s 25ms/step - batch: 124.5000 - size: 1.0000 - loss: 0.1131 - acc: 0.9600 - val_loss: 0.1931 - val_acc: 0.9303		
AI Model used and training results based on created vitiligo dataset.		
Model: "sequential"		
Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 40, 40, 3)	0
conv2d (Conv2D)	(None, 78, 78, 32)	896
max_pooling2d (MaxPooling2D	(None, 39, 39, 32)	0)
conv2d_1 (Conv2D)	(None, 37, 37, 32)	9248
max_pooling2d_1 (MaxPooling	(None, 18, 18, 32)	0
2D)		
conv2d_2 (Conv2D)	(None, 16, 16, 32)	9248
max_pooling2d_2 (MaxPooling	(None, 8, 8, 32)	0
2D)		
flatten (Flatten)	(None, 2048)	0
dense (Dense)	(None, 128)	262272
dense_1 (Dense)	(None, 2)	258
Total params: 281,922		
Trainable params: 281,922		

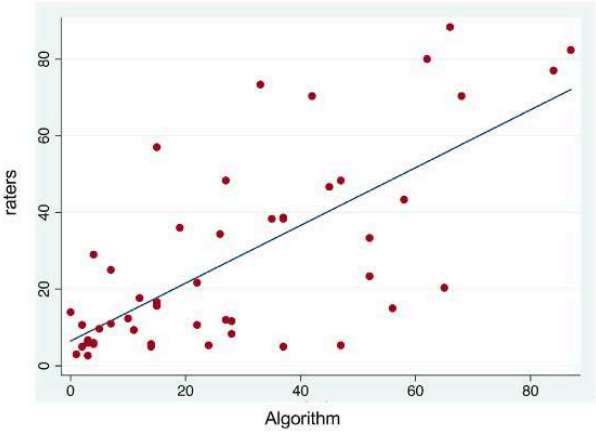
(*continued*)

Continued

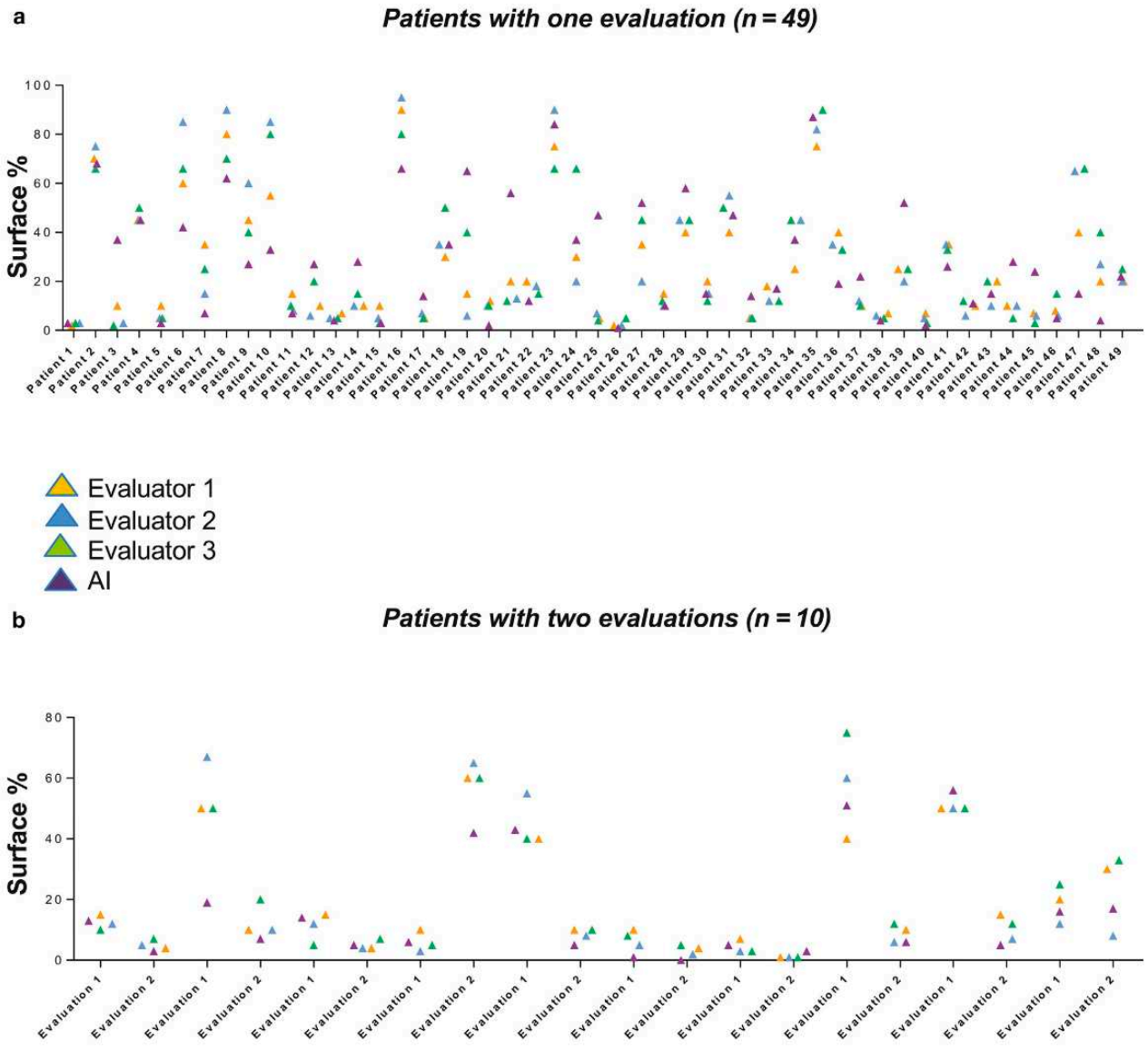
Non-trainable params: 0
Train on 250 steps, validate on 63 steps
Epoch 1/30
249/250 [==============================>.] - ETA: 0s - batch: 124.0000 - size: 1.0000 - loss: 0.3801 - acc: 0.8389
250/250 [==============================] - 7s 26ms/step - batch: 124.5000 - size: 1.0000 - loss: 0.3824 - acc: 0.8387 - val_loss: 0.3129 - val_acc: 0.8796
Epoch 30/30
250/250 [==============================] - 6s 25ms/step - batch: 124.5000 - size: 1.0000 - loss: 0.1131 - acc: 0.9600 - val_loss: 0.1931 - val_acc: 0.9303
AI Model used and training results based on created vitiligo dataset.
Model: "sequential"

Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 160, 160, 3)	0
conv2d (Conv2D)	(None, 78, 78, 32)	896
max_pooling2d (MaxPooling2D	(None, 39, 39, 32)	0)
conv2d_1 (Conv2D)	(None, 37, 37, 32)	9248
max_pooling2d_1 (MaxPooling	(None, 18, 18, 32)	02D)
conv2d_2 (Conv2D)	(None, 16, 16, 32)	9248
max_pooling2d_2 (MaxPooling	(None, 8, 8, 32)	02D)
flatten (Flatten)	(None, 2048)	0
dense (Dense)	(None, 128)	262272
dense_1 (Dense)	(None, 2)	258

Total params: 281,922
Trainable params: 281,922
Non-trainable params: 0
Train on 250 steps, validate on 63 steps
Epoch 1/30
249/250 [==============================>.] - ETA: 0s - batch: 124.0000 - size: 1.0000 - loss: 0.3801 - acc: 0.8389
250/250 [==============================] - 7s 26ms/step - batch: 124.5000 - size: 1.0000 - loss: 0.3824 - acc: 0.8387 - val_loss: 0.3129 - val_acc: 0.8796
Epoch 30/30
250/250 [==============================] - 6s 25ms/step - batch: 124.5000 - size: 1.0000 - loss: 0.1131 - acc: 0.9600 - val_loss: 0.1931 - val_acc: 0.9303

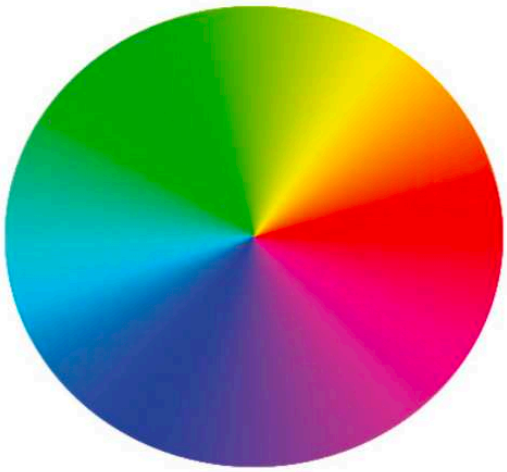


Supplementary Figure S1. Correlation analysis between the mean score of all physicians and the AI. Scatter plot pairs of numerical data: mean of physicians' evaluations (on the y-axis) and scoring of the AI system (on the x-axis). The regression line shows the pattern of data. AI, artificial intelligence.

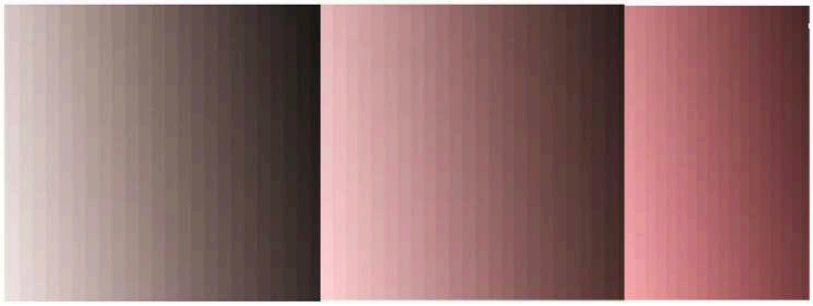


Supplementary Figure S2. Scores of evaluators and AI. Panels show the (**a**) result of physicians' evaluations and AI for all patients with only one evaluation (n = 49) and (**b**) for patients with two evaluations at two time points (n = 10). The evaluation was done by three physicians (red, gray, and yellow dots) and AI (blue dots). AI, artificial intelligence.

a



b



Supplementary Figure S3. HSV color space and vitiligo skin values. The panel shows the HSV color wheel and vitiligo skin color ranges. HSV, hue, saturation, value/luminance.

UNet Implementation

Cost Functions and UNet

```
def dice_coef(y_true, y_pred):
    y_true_f = K.flatten(y_true)
    y_pred_f = K.flatten(y_pred)
    intersection = K.sum(y_true_f * y_pred_f)
    return (2.0 * intersection + 1.0) / (K.sum(y_true_f) + K.sum(y_pred_f) + 1.0)

def dice_coef_loss(y_true, y_pred):
    return 1-dice_coef(y_true, y_pred)

def jacard_coef(y_true, y_pred):
    y_true_f = K.flatten(y_true)
    y_pred_f = K.flatten(y_pred)
    intersection = K.sum(y_true_f * y_pred_f)
    return (intersection + 1.0) / (K.sum(y_true_f) + K.sum(y_pred_f) - intersection + 1.0)

def jacard_coef_loss(y_true, y_pred):
    return 1-jacard_coef(y_true, y_pred)

def Attention_ResUNet( dropout_rate=0.0, batch_norm=True):
    "Residual UNet, with attention"'
    # network structure
    FILTER_NUM = 64 # number of basic filters for the first layer
    FILTER_SIZE = 3 # size of the convolutional filter
    UP_SAMP_SIZE = 2 # size of upsampling filters
    # network structure
    FILTER_NUM = 32  # number of basic filters for the first layer
    FILTER_SIZE = 3  # size of the convolutional filter
    DOWN_SAMP_SIZE = 2  # size of pooling filters
    UP_SAMP_SIZE = 2  # size of upsampling filters
    INPUT_CHANNEL = 3  # 1-grayscale, 3-RGB scale
    OUTPUT_MASK_CHANNEL = 1
    INPUT_SIZE = 128
    '"Residual UNet construction, with attention to the gate
    convolution: 3*3 SAME padding
    pooling: 2*2 VALID padding
    upsampling: 3*3 VALID padding
    final convolution: 1*1
    :param dropout_rate: FLAG & RATE of dropout.
            if < 0 dropout cancelled, if > 0 set as the rate
    :param batch_norm: flag of if batch_norm used,
            if True batch normalization
    :return: model
    '"
    # input data
    # dimension of the image depth
    inputs = layers.Input(
        (INPUT_SIZE, INPUT_SIZE, INPUT_CHANNEL), dtype="float32")
    axis = 3
    # Downsampling layers
    # DownRes 1, double residual convolution + pooling
    conv_128 = res_conv_block(inputs, FILTER_SIZE, FILTER_NUM, dropout_rate, batch_norm)
```

```
    pool_64 = layers.MaxPooling2D(pool_size=(2,2), data_format="channels_last")(conv_128)
    # DownRes 2
    conv_64 = res_conv_block(pool_64, FILTER_SIZE, 2*FILTER_NUM, dropout_rate, batch_norm)
    pool_32 = layers.MaxPooling2D(pool_size=(2,2), data_format="channels_last")(conv_64)
    # DownRes 3
    conv_32 = res_conv_block(pool_32, FILTER_SIZE, 4*FILTER_NUM, dropout_rate, batch_norm)
    pool_16 = layers.MaxPooling2D(pool_size=(2,2), data_format="channels_last")(conv_32)
    # DownRes 4
    conv_16 = res_conv_block(pool_16, FILTER_SIZE, 8*FILTER_NUM, dropout_rate, batch_norm)
    pool_8 = layers.MaxPooling2D(pool_size=(2,2), data_format="channels_last")(conv_16)
    # DownRes 5, convolution only
    conv_8 = res_conv_block(pool_8, FILTER_SIZE, 16*FILTER_NUM, dropout_rate, batch_norm)

    # Upsampling layers
    # UpRes 6, attention gated concatenation + upsampling + double residual convolution
    gating_16 = gating_signal(conv_8, 8*FILTER_NUM, batch_norm)
    att_16 = attention_block(conv_16, gating_16, 8*FILTER_NUM)
    up_16 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE),
data_format="channels_last")(conv_8)
    up_16 = layers.concatenate([up_16, att_16], axis=axis)
    up_conv_16 = res_conv_block(up_16, FILTER_SIZE, 8*FILTER_NUM, dropout_rate, batch_norm)
    # UpRes 7
    gating_32 = gating_signal(up_conv_16, 4*FILTER_NUM, batch_norm)
    att_32 = attention_block(conv_32, gating_32, 4*FILTER_NUM)
    up_32 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE),
data_format="channels_last")(up_conv_16)
    up_32 = layers.concatenate([up_32, att_32], axis=axis)
    up_conv_32 = res_conv_block(up_32, FILTER_SIZE, 4*FILTER_NUM, dropout_rate, batch_norm)
    # UpRes 8
    gating_64 = gating_signal(up_conv_32, 2*FILTER_NUM, batch_norm)
    att_64 = attention_block(conv_64, gating_64, 2*FILTER_NUM)
    up_64 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE),
data_format="channels_last")(up_conv_32)
    up_64 = layers.concatenate([up_64, att_64], axis=axis)
    up_conv_64 = res_conv_block(up_64, FILTER_SIZE, 2*FILTER_NUM, dropout_rate, batch_norm)
    # UpRes 9
    gating_128 = gating_signal(up_conv_64, FILTER_NUM, batch_norm)
    att_128 = attention_block(conv_128, gating_128, FILTER_NUM)
up_128 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE),
data_format="channels_last")(up_conv_64)
    up_128 = layers.concatenate([up_128, att_128], axis=axis)
    up_conv_128 = res_conv_block(up_128, FILTER_SIZE, FILTER_NUM, dropout_rate, batch_norm)

    # 1*1 convolutional layers
    conv_final = layers.Conv2D(OUTPUT_MASK_CHANNEL, kernel_size=(1,1))(up_conv_128)
    conv_final = layers.BatchNormalization(axis=axis)(conv_final)
    conv_final = layers.Activation('sigmoid')(conv_final)  #Change to softmax for multichannel

    # Model integration
    model = models.Model(inputs, conv_final, name="AttentionResUNet")
    return model
```

Prediction

```
model, current_epoch = create_model()
img = load_image_data("./Archive1/Skin/img-rev_0-0-BEZ_MA 2 (17).jpg.tiff")
pred = model.predict(np.array([img]))
keras.utils.save_img("./Archive1/Prediction/img-rev_0-0-BEZ_MA 2 (17).jpg.tiff", pred[0] * 255,
scale=False)
```

CV

11/2020 - 05/2024 PhD Student University of Bordeaux

05/2021 – 05/2022: Senior Expert, Development of Digitalisation Projects based on ServiceNow, Arctive AG

11/2020 – 04/2021: Consultant, Development of AI-based System for Medical Image Analysis, Freelance Consulting

05/2020 – 11/2020: Senior Consultant, Development and consulting on SIEM based on IBM QRadar, Serima Consulting

02/2007 – 04/2020: Senior IT Specialist, Project Management and development for SOX-based Incident, Problem, Change-, Release-, and Deployment system, IBM

11/2005 - 02/2007: Senior Consultant, Managing a Centralized Service Desk Application for T-Systems Germany, Devoteam Schweiz AG

06/2002 - 11/2005: Application Consultant, International project management and coordination, T-Systems Schweiz AG

04/2001 - 04/2002: Technical Product Integration Manager – EMEA, Technical and Business Development support for Product Partners, Peregrine Systems

05/1999 - 04/2001: IT Manager, International Project management and roll out for ENCODEX service in Europe, ENCODEX International

05/1997 - 05/1999: Application Consultant, Technical point of contact for interface development, Remedy Corporation

1995: Education of Flight Assistance, Condor Airline

2014: Patent – Management of Configuration Data Structures in Multi-Layer Data Models

1997: Publication – Dirk Hillmer, "VirTeam: Visibility in Virtual Groups”

1997: Publication – Dirk Hillmer, "VirTeam Tool”

1997: ASME Design Engineering Technical Conferences, Sacramento (CA)

Diplom Informatiker in Informatik, Technische Universität Darmstadt,
Minor: BWL, Diplom
Thesis: Collaboration in Virtual Teams, 1996