UNIVERSITY OF PIRAEUS
School of Information and Communication Technologies
Department of Informatics

Thesis

Thesis Title:	Detecting Interstellar Bubbles in Images using CNNs
Τίτλος Διατριβής:	Ανίχνευση Κοσμικών Φυσαλίδων σε Εικόνες με χρήση ΣΝΔ
Student's name-surname:	Christina-Styliani Andrianopoulou
Father's name:	Evangelos
Student's ID No:	Π11004
Supervisor:	Dionysios Sotiropoulos, Assistant Professor

June 2025/ Ιούνιος **2025**

Abstract

Early-type stars create stellar-wind bubbles often with irregular shapes, causing their detection to be both time-consuming and prone to human biases. In this dissertation, we present a program to create and train a Convolutional Neural Network (CNN) to detect these stellar objects which are termed Interstellar Bubbles by astronomers, using image detection. These bubbles have been areas of interest for a while as they have been observed to be a hosting environment for star formation and because they are visually distinct. Recent advances in deep learning, particularly CNNs, have allowed an automated approach to detect these structures with improved efficiency and accuracy. Here we present a CNN model with a Residual U-Net architecture, trained on the Milky Way Project data from Spitzer telescope, that identifies bubbles with 91% accuracy and nearly 1% false positive rate and outputs their location and shape parameters.

Keywords: ISM Bubbles, Deep Learning, Convolutional Neural Network, Image Segmentation

Περίληψη

Αστέρες που βρίσκονται στο πρώιμο στάδιο εξέλιξης τους δημιουργούν φυσαλίδες αστρικών ανέμων με συχνά ακανόνιστα σχήματα, με αποτέλεσμα η ανίχνευσή τους να είναι χρονοβόρα και επιρρεπής σε ανθρώπινη μεροληψία. Σε αυτή την εργασία παρουσιάζουμε έναν αλγόριθμο για τη δημιουργία και εκπαίδευση ενός Συνελικτικού Νευρωνικού Δικτύου (ΣΝΔ) που εντοπίζει αυτά τα αστρικά αντικείμενα που οι αστρονόμοι ορίζουν ως Κοσμικές Φυσαλίδες, χρησιμοποιώντας αναγνώριση εικόνας. Οι φυσαλίδες αυτές έχουν αποτελέσει εδώ και καιρό αντικείμενο έρευνας καθώς έχουν παρατηρηθεί να δρουν ως περιβάλλον που φιλοξενεί σχηματισμούς νέων αστέρων αλλά και επειδή είναι οπτικά διακριτές. Οι πρόσφατες εξελίξεις στη βαθιά μάθηση, ιδιαίτερα στα ΣΝΔ, έχουν ευνοήσει μια πιο αυτοματοποιημένη προσέγγιση για την ανίχνευση αυτών των δομών με βελτιωμένη αποτελεσματικότητα και ακρίβεια. Στη παρούσα εργασία παρουσιάζουμε ένα μοντέλο ΣΝΔ με αρχιτεκτονική υπολειπόμενου U-Net νευρωνικού δικτύου, εκπαιδευμένο με βάση τα δεδομένα του Milky Way Project από το τηλεσκόπιο Spitzer, πετυχαίνοντας ανίχνευση φυσαλίδων με ακρίβεια 91% και ψευδοθετικό ποσοστό σχεδόν 1%, ενώ εξάγει την τοποθεσία και τις σχηματικές παραμέτρους τους.

Λέξεις-κλειδιά: Κοσμικές Φυσαλίδες, Βαθιά Μάθηση, Συνελικτικό Νευρωνικό Δίκτυο, Κατάτμηση Εικόνας

Πρόλογος-Ευχαριστίες

Η παρούσα πτυχιακή εργασία εκπονήθηκε στο πλαίσιο της φοίτησης στο Τμήμα Πληροφορικής του Πανεπιστημίου Πειραιώς, με σκοπό την απόκτηση του πτυχίου της Σχολής Τεχνολογιών Πληροφορικής και Επικοινωνιών.

Αντικείμενο της εργασίας αποτελεί η ανάπτυξη και αξιολόγηση ενός αλγορίθμου εντοπισμού φυσαλίδων σε αστρονομικά δεδομένα με τη χρήση Συνελικτικών Νευρωνικών Δικτύων. Η μελέτη επικεντρώνεται στον σχεδιασμό και την εκπαίδευση ενός μοντέλου U-Net για την κατάτμηση εικόνων, την επιλογή του μοντέλου με κριτήριο την απόδοση διαφορετικών συναρτήσεων απώλειας, καθώς και την εξαγωγή των ζητούμενων παραμέτρων.

Σε αυτό το σημείο θα ήθελα να εκφράσω τις ευχαριστίες μου στον επιβλέποντα μου, Δρ. Διονύσιο Σωτηρόπουλο, για την καθοδήγηση, τη στήριξη και την πολύτιμη βοήθεια καθ' όλη την πορεία της εργασίας, όπως επίσης και τη Δρ. Elizabeth Watkins, για την συνδρομή της στον έλεγχο της ορθότητας της αστρονομικής πλευράς αυτής της μελέτης.

Τέλος, ευχαριστώ την οικογένειά μου και τους φίλους μου για την ενθάρρυνση και την υποστήριξη κατά τη διάρκεια της εκπόνησης.

Introduction

In Section 1, we outline the background of this work, including the basic astronomical and computing concepts needed to understand this work. We begin by describing the nature of interstellar bubbles along with how they have been observed to form over the years through stellar feedback processes (Churchwell et al., 2006; Deharveng et al., 2010; Geen et al., 2021) and what information they provide astronomers regarding their surrounding environments (Peeters et al., 2004). We then highlight the astrophysical motivation and the importance of accurate bubble identification in understanding the galactic structure and star formation. Section 2 presents a review of the existing literature on interstellar bubbles and any relevant work that attempts to automate the bubble detection process. The first part of Section 2 focuses on the image analysis techniques used in a similar context while the second part delves into the deep learning techniques used in image processing, presenting recent CNN architectures and training strategies used in object detection and classification. In Section 3, we introduce the Spitzer dataset and the MWP catalog containing the labels used for training and evaluation. Here, we outline all the preprocessing steps applied prior to model training and all the techniques used to improve data quality and consistency. Section 4 provides a summary of the Deep Learning Architectures used in this project and how these were implemented in our CNN model, along with the metrics used to evaluate our model's performance. In Section 5, we list our results and discuss the findings of our work while assessing our model's ability to detect bubbles on unseen Spitzer datasets.

Contents

1 Context and Background

Stars, generate stellar winds that produce irregular and elliptic shapes often referred to as bubbles in 2D. They are hot, ionised cavities surrounded by a shell of cooler gas and dust, which extend several parsecs into the interstellar medium and have been observed to be sites of star formation. Identifying those areas has proven a relatively tedious and time-consuming process due to the vast amount of data, leading experts to employ citizen-science methods to speed up the task, such as the Milky-Way Project (*Zooniverse.org*[1]). In this project, citizen scientists were asked to detect and classify various stellar objects that were later cross-examined by astronomers, resulting in a catalog listing the statistical mean values of the identified bubbles, together with a few new items of interest for the astronomical community, such as yellow-balls and bow shocks. Yellow-balls are hyper-compact HII regions that have not expanded and serve as indicators of early, massive star formation, often found along the rims of large bubbles. Bow shocks, on the other hand, are arc-shaped features that appear red in IR images due to the presence of warmer dust and ionized gas. These objects are separate from bubbles and serve different research purposes.

1.1 Interstellar-wind Bubbles

Our main focus revolves around interstellar bubbles, which are a byproduct of massive stars during their main-sequence and post-main-sequence phases. As these stars emit ultraviolet radiation and eject stellar winds, they carve out large cavities in the surrounding gas. These bubbles are actually dynamic, evolving structures. Their interiors are filled with hot, ionized gas, while the surrounding dense shells often become sites of further star formation. This occurs as the expanding shock waves compress nearby molecular clouds, initiating the collapse of gas into new stars, as described from the collect-and-collapse model, or C&C for short, defined in Elmegreen and Lada, 1977.

[1]Zooniverse.org: https://www.zooniverse.org/projects/povich/milky-way-project

The detection and characterization of interstellar bubbles are highly dependent on the resolution of observations, which varies with the distance of the target galaxy. In nearby galaxies, high-resolution imaging reveals smaller, well-defined bubbles primarily shaped by stellar winds from massive stars and supernova-driven shock waves. In contrast, in more distant galaxies ($\sim$ 30 Mpc away) exhibit larger, more diffuse bubbles, often dominated by radiation pressure, as individual structures blend together due to lower angular resolution (Kim et al., 2023; Watkins et al., 2023).

Infrared observations play a crucial role in tracing bubble properties. The 8 μm band, observed by Spitzer's Galactic Legacy Infrared Mid-Plane Survey Extraordinair (GLIMPSE) and James Webb Space Telescope 's 7.7μm, highlights the presence of polycyclic aromatic hydrocarbons (PAHs), which serve as sensitive tracers of gas density and molecular or ionized gas (Peeters et al., 2004; Salama, 2008). However, PAH emission can be suppressed in harsh radiation fields, making it an environmental-dependent indicator. The 24 μm band complements this by detecting warm dust heated by embedded star formation, in both Milky Way and nearby galaxies, where individual star-forming regions can be resolved (Churchwell et al., 2006; Deharveng et al., 2010).

In addition to the above, the molecular composition of the interstellar medium (ISM) provides insight into bubble environments. PAH molecules, which vibrate and twist under stellar UV light, trace regions with finer dust ($\sim$100 K). In contrast, ^{12}CO emission, a common tracer of molecular gas due to its low-energy rotational transitions effective around $\sim$20 K, predominantly traces cold, shielded regions. However, in the presence of strong UV fields—such as those near bubble rims—CO can be selectively photodissociated, making it an incomplete tracer of molecular gas in UV-irradiated environments (Bolatto et al., 2013; Tielens, 2005; Visser et al., 2009).

This limitation underlines the importance of complementary tracers, such as dust emission (including PAHs), which remain observable even where CO has been destroyed. By combining multiple wavelengths, we can differentiate between actively star-forming regions and more dormant molecular clouds. These observational insights signify the need to develop a CNN for this task, as the varying bubble characteristics and emission profiles across different environments shape the features that the model learns to identify.

In the present work, we strictly focus on automating the process of identifying PAH shells using the 8 μm wavelength data, to simplify the process in our limited resourced systems.

1.2 Convolutional Neural Networks

Scientists have evolved their methods to take advantage of technological advances to handle the increasing volume and complexity of astronomical data. From manual extraction of features for training SVMs to distinguish supernova remnants using CO emission (Xu et al., 2020b) and identify molecular outflows (Xu et al., 2020a), to Machine Learning (ML) and Deep Learning (DL) applications.

Beaumont et al., 2014 developed BRUT, a Random Forest algorithm designed to identify bubbles in IR images from Spitzer Space Telescope, which is composed of various decision trees in an effort to automate bubble detection, resulting in a bubble distinction and classification rate similar to an expert. While ML uses algorithms to parse the data and eventually learn from them relevant patterns, an Artificial Neural Network, or simply Neural Network, replicates the architecture of the human brain, consisting of interconnected nodes (or neurons) organized into layers that process and transmit information to learn complex representations of the data (LeCun et al., 2015). Among them, CNNs are particularly suited for image analysis, as they extract patterns from raw pixel data (Domínguez Sánchez et al., 2020).

Deep Convolutional Neural Networks (DCNNs) neurons self-optimize through learning. Using backpropagation, CNNs fine-tune their filters and weights during their training, making them the perfect candidates for pattern recognition tasks within images, as they effectively detect features such as textures, shapes, and-or edges (Dieleman et al., 2015;

Geirhos et al., 2022).

1.3 Importance of the Problem

Detecting and analyzing interstellar bubbles is crucial to understanding star formation and the dynamics that dominate the interstellar medium, as they strongly influence the chemical composition, density distribution, and magnetic properties of the ISM. Traditional approaches to bubble identification rely heavily on visual inspection of the infrared and radio observations and manual cataloging from experts, similar to the work performed using data from Galactic Legacy Infrared Mid-Plane Survey Extraordinaire (GLIMPSE) at 8 µm and MIPSGAL Survey at 24 µm, as well as Milky Way Project (MWP) surveys (Churchwell et al., 2006; Simpson et al., 2012). Performing this demanding task on a CNN trained to handle noisy data and recognize faint shapes, allows us to handle huge volumes of data more efficiently, in a shorter time. Furthermore, Geirhos et al., 2022 shows that CNNs are prone to be "texture-biased", which is also demonstrated in Islam et al., 2021, and more specifically for ImageNet-trained CNNs. As PAH emission creates distinctive textured rims, we are confident that CNNs can detect them more effectively.

Although previous efforts such as BRUT (Beaumont et al., 2014) and CASI(Van Oort et al., 2019) have automated aspects of bubble detection, few studies have focused specifically on the detailed morphological characterization of bubbles, particularly using CNNs trained on mid-infrared data to extract the shape properties of a bubble. Here, we attempt to offer a solution by expanding the scope of this project to further include image segmentation. This approach enables us to obtain additional morphological properties of each bubble, in addition to its location. We specifically aim to extract shape parameters such as its effective radius, eccentricity and position angle.

2 Literature Review

2.1 Image Analysis Techniques

Depending on the form of data available, whether these are radio data or infrared images, scientists have performed a variety of different methods. A visual identification approach, is used in a study carried out on NGC 628 data, where bubbles are identified using their shell-like characteristics in a combination of MIRI F770W observations, supplied by JWST and Spitzer-GLIMPSE 8.0 μm data (Watkins et al., 2023). Another method includes the creation and analysis of color composite images from different wavelengths, with Spitzer-GLIMPSE 8.0 μm and Spitzer-MIPSGAL 24.0 μm data, to compile images containing ionized gas and dust emission. MWP also provided similar RGB composites to citizen scientists for bubble identification and cataloging (Kendrew et al., 2012; Simpson et al., 2012).

These RGB composite infrared images compose the training set of BRUT, a machine learning algorithm based on Random Forests. BRUT analyzes features extracted from preprocessed images and utilizes intensity clipping and normalization to classify images on whether they contain a bubble or not (Beaumont et al., 2014). Other image processing techniques such as dilation, erosion and gradient are used to extract regions with smooth boundaries and are applied to detect bubble-shaped structures, like supernova shells, by identifying regions of density decrease (Van Oort et al., 2019). The same algorithm was re-trained with synthetic bubble images that improved its performance in identifying certain types of bubble(Nishimoto et al., 2025).

2.2 Deep Learning Techniques

Deep Learning (DL) has been a breakthrough in data analysis, due to its ability to manipulate large amounts of data. Despite astronomers initially being skeptical about incorporating NN into research, DL has become an increasingly valuable tool in astronomical analysis, where data volume is growing exponentially-doubling approximately every 16 months, according to Smith and Geach, 2023. Instead of manually programming a system to identify feature patterns, scientists rely on DL models to learn these patterns directly from the data. A remarkable example of DL is the CNN developed by Dieleman et al., 2015, for galaxy morphology classification, which handles rotation invariance by augmenting its dataset with rotated images of the input images. DL has also proven useful for other tasks in astronomy, such as denoising astronomical images with U-Net architectures Vojtekova et al., 2021 and enhancing the resolution and detail of galaxy images using Generative Adversarial Networks (GANs) Schawinski et al., 2017.

A notable deep learning-based approach to shell detection is CASI (stands for Convolutional Approach for Shell Identification), developed byVan Oort et al., 2019. CASI utilizes a CNN trained on labeled Spitzer 8 μm images from the GLIMPSE survey to classify regions containing ^{12}CO shell-like structures. Unlike BRUT, which relies on feature extraction and the Random Forest method, CASI directly learns spatial and morphological features from pixel data. It performs well in identifying bubble morphologies across varying brightness levels and scales well to large datasets. CASI represents an early application of CNNs to mid-infrared data in order to isolate and characterize interstellar feedback structures.

For bubble detection, DL involves training a CNN model on labeled examples of bubbles, allowing the model to generalize and recognize similar structures in unseen images. Once trained, a model can identify interstellar bubbles across diverse environments and varying observational conditions.

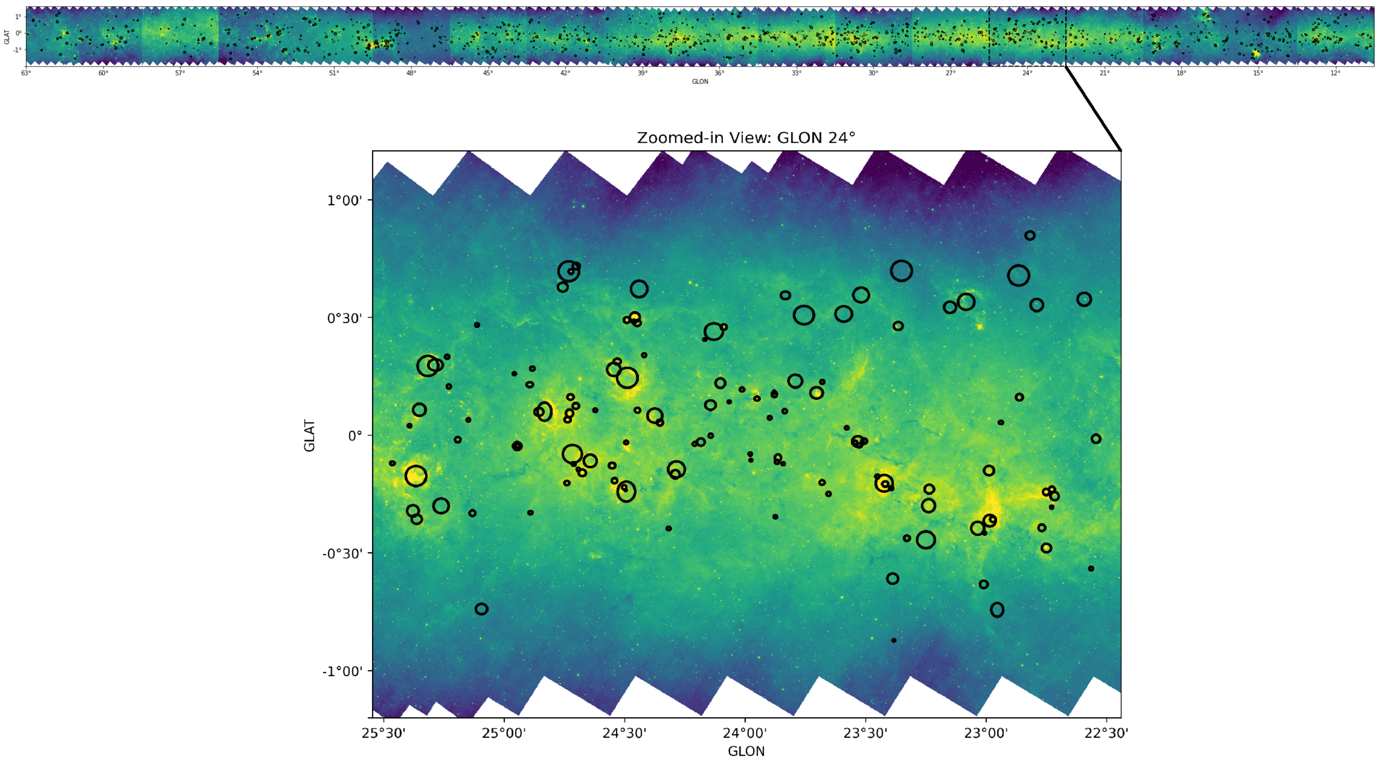

FIGURE 1: Top Panel: Compiled mosaic of the entire GLIMPSE survey at 24μm along the Galactic plane FITS files with detected bubbles. Bottom Panel: Zoom-in of the GLIMPSE survey at l=24°.

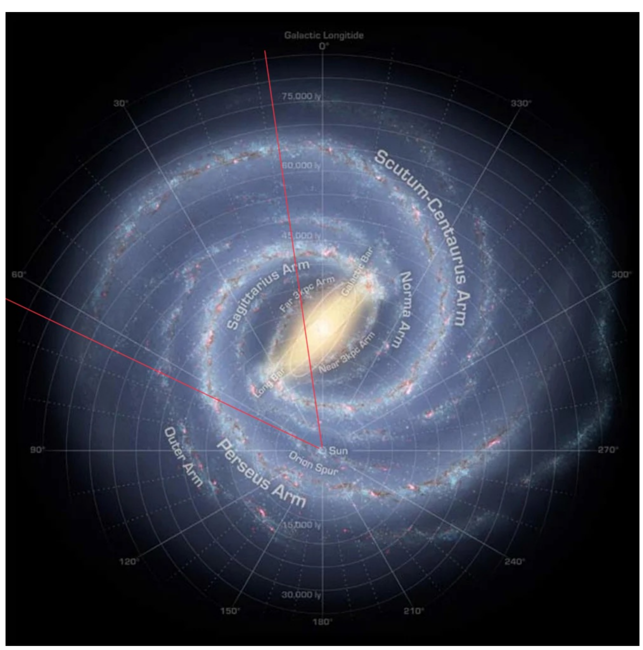

FIGURE 2: Artistic Impression of a bird's eye view of the Milky Way

3 Dataset

The primary dataset used to train our CNN model consists of 8 μm infrared images from the Spitzer Space Telescope (SST), more specifically from the GLIMPSE survey. SST was the third space telescope launched under NASA's Great Observatories program to collect data in the infrared spectrum and operated until in early 2020. These observations inspect and identify various distinct features, such as the infrared emission from polycyclic aromatic hydrocarbons (PAHs) at bubble rims, which glow brightly in infrared wavelengths or synthetic 12CO emissions used in the CASI model (Van Oort et al., 2019). Our data set consists mainly of the FITS files provided from the GLIMPSE survey. We select non-overlapping GLIMPSE snapshots with a Galactic Longitude range of 10.5° to 64.5°, each spanning approximately 3° in width, as shown in Figure 1. These regions constitute the training and validation sets. Files from this survey lying outside this range are reserved for testing, due to minor differences in formatting. A visual representation of the selected range projected on the Milky Way is depicted in Figure 2.

The Spitzer Space Telescope 8 μm images from the GLIMPSE survey[2] and the Milky Way Project catalog[3] serve as the primary data sources for our study.

3.1 Preprocessing

The main idea is to follow a procedure similar to BRUT's training by creating astronomical image cutouts centered on infrared bubbles to train our CNN model to segment and trace these structures. While the Milky Way Project data set provides an ample amount of well-labeled bubbles, we still face the challenge of working with a relatively small sample for training a deep learning model.

We define positive samples as cutouts centered on MWP-identified bubbles with complete shape parameters. As shown in Table 1, not all catalogue objects satisfy the required

[2]Available at https://irsa.ipac.caltech.edu/data/SPITZER/GLIMPSE/images/I/1.2_mosaics_v3.5/
[3]Available at https://github.com/skendrew/milkywayproject_triggering/.../mwp-all-bubbles-dr1.csv

shape criteria, like effective radius and eccentricity, and several of the remaining items fall outside of our study range. Hence, before preprocessing, we first remove all elements that exist outside the GLON 10.5°-64.5° window as well as any items that contain missing or undefined values in the effective radius and eccentricity fields, to secure compliance with the shape requirements. This filtering step eliminates 27.5% of the catalog due to insufficient morphological information.

Another limitation that arises due to processing power restrictions is that we do not consider bubbles larger than our 256×256 bounding box. These dimensions are specifically selected as they do not significantly reduce the available dataset while still allowing the majority of small bubbles to fit in the training box.

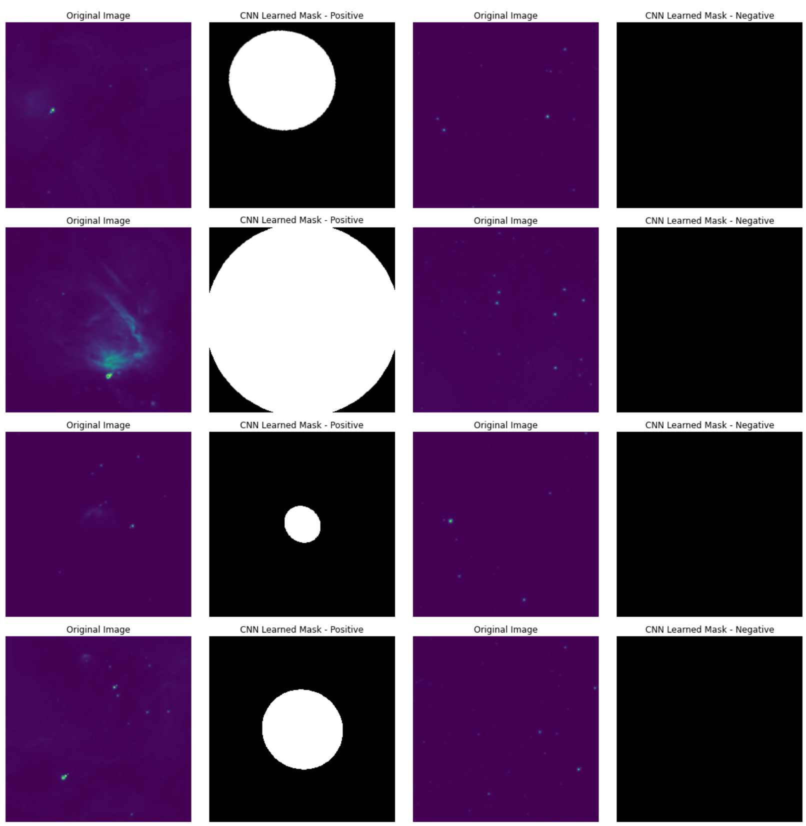

FIGURE 3: Representative training samples used for CNN segmentation. Each row presents a pair of input cutouts: a positive example containing a bubble (left) and a negative example without a bubble (right). For each, the original Spitzer 8μm image is shown on the left, and the corresponding ground truth mask used for training is shown on the right. The masks for positive samples depict the annotated bubble regions, while those for negatives are entirely blank.

During preprocessing, all sky coordinates are converted to image pixel coordinates and indexed, so that they can be safely retrieved and reverse mapped later on. For each cutout extracted that contains a positive image, we create a corresponding binary mask of the ellipse, while the rest of the cutouts are also sampled as negative (non-bubble) regions. Figure 3 illustrates this, with pairs of positive images and their masks depicted on the left and examples of negative associated pairs on the right.

Considering that some bubble formations are quite elaborate and that our dataset is fairly imbalanced, as images containing bubbles comprise roughly 20% of the total, it is necessary to perform several data augmentation methods to ensure that our model can detect those intricate shapes. Using the `ImageDataGenerator`, we apply 20° rotations, horizontal and vertical flips, and minor shifts to each original bubble image, effectively expanding the positive sample set fourfold. This augmentation results in a near 1:1 ratio of positive to negative samples.

The entire dataset is split into training, validation, and test sets using a 60-20-20 ratio. All sets are then normalized, using Min-Max rescaling followed by standard normalization with the mean and standard deviation of the training set. This ensures consistent input ranges, improves the model's convergence during training and prevents data leakage from validation and test sets.

4 Methodology

We train our model using the Adam optimizer with a learning rate of 10^{-4}. The training is implemented from scratch, initializing weights randomly without relying on any pretrained backbone. The dataset is split into training, validation, and test sets using a 60-20-20 ratio. The training and validation sets are used to optimize the model and tune its parameters, while the test set is kept completely separate for the final performance evaluation.

A batch size of 16 was empirically selected and the model is trained for a maximum of 100 epochs. To avoid overfitting and reduce training time, we implement early stopping with a patience of a reasonable amount of epochs(5), while monitoring the validation loss during training.

As described in the preprocessing section, data augmentation is applied only to the training set, therefore increasing its effective size and helping the model generalize better to diverse bubble shapes and orientations. This includes rotations, axis flips, and small translations applied using the `ImageDataGenerator` utility.

The primary metric used for evaluation is the Dice coefficient, which directly reflects the spatial overlap between the predicted and ground truth segmentation masks. This is particularly appropriate for our application due to the highly imbalanced nature of the dataset, where the majority of each image consists of background pixels.

In post-processing, the predicted binary masks are converted into labeled regions using connected-component analysis. For each predicted bubble, we extract morphological properties such as effective radius, eccentricity, and position angle by fitting ellipses to the detected masks.

4.1 U-Net Architecture

Following the technological advancements in the Deep Learning field, we develop and train a Convolutional Neural Network to perform image segmentation with the purpose of identifying and tracing the shape of interstellar bubbles. Typically, a CNN works by progressively reducing an image to its most significant features through a series of convolutions and non-linear activations. In this work, we employ the Residual U-Net architecture, or ResUNet, which is notable for addressing the vanishing gradient problem commonly encountered in deep CNNs. ResUNet combines the encoder–decoder framework of U-Net enriched with residual blocks, enhancing both training stability and feature propagation.

The U-Net architecture became widely recognized after its breakthrough in the medical field through the work of Ronneberger et al., 2015, where they achieved high accuracy in biomedical image segmentation tasks. In the context of astronomy, U-Net has been adapted by Vojtekova et al., 2021 to implement ASTRO U-Net, a CNN trained to denoise and enhance astronomical observations, producing results that resemble images taken with longer exposure time. For our study, U-Net was selected mainly because it is well-suited for relatively small datasets like ours. It works by skipping connections between down-sampling and up-sampling paths to preserve spatial information and high-level features. It can also expand the image back to its original size to precisely learn where the information is located. For bubble detection, this proves essential in identifying the characteristic PAH emission at bubble rims and distinguishing ISM bubble cavities from other complex ISM structures. Figure 4 presents our ResU-Net architecture used for the image segmentation task.

To improve generalization, we enhance our model by adding residual blocks. These blocks create "shortcut" connections that allow information to pass forward directly, bypassing intermediate layers and matching the dimensions of the destination layer. In effect, residual connections help the NN maintain stability during backpropagation and support deeper architectures, while also improving feature reuse and learning efficiency. This helps prevent the model from struggling to update the weights, while we also improve the process of learning.

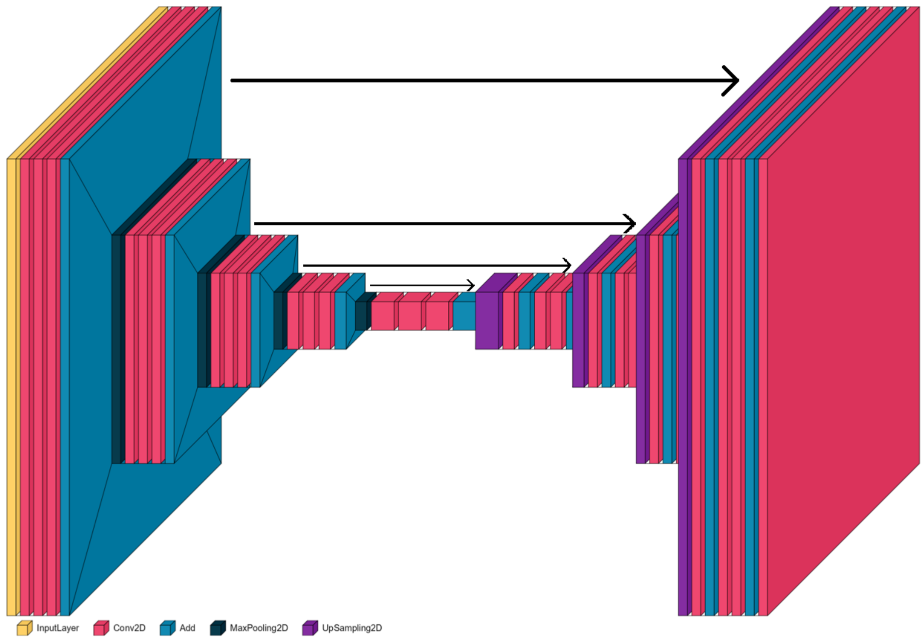

FIGURE 4: Architecture of the CNN used for bubble segmentation. The model follows a U-Net structure composed of a contracting path (left) for feature extraction and an expansive path (right) for spatial reconstruction. Each block includes convolutional layers (red), downsampling via max pooling (blue), and upsampling layers (purple). Skip connections (black arrows) merge encoder and decoder features at corresponding spatial scales, preserving fine details during upsampling. The model takes 256×256 Spitzer 8μm cutouts as input and outputs binary segmentation masks identifying potential interstellar bubbles.

4.2 Loss function selection

An important part of our model is the selection of a suitable loss function for the segmentation task. Although our pixels are binary classified, we dismiss Binary Crossentropy (BCE) as suitable candidate, due to the significant class imbalance: the majority of pixels belong to the background class -or non-bubble class- rather than the bubble structure. This could lead the model to overfit toward the majority class and miss the bubble boundaries entirely.

Other potential candidates we consider include Focal, Dice and Boundary loss functions. Selecting the appropriate loss function depends on what we want the model to emphasize. Focal loss is designed to focus on difficult-to-classify pixels, down-weighing easy training examples, and assisting in rare bubbles' detection that we would otherwise ignore. Dice loss, on the other hand, is effective in handling class imbalance as it measures the overlap of predicted and ground truth masks and uses that to compare, which aligns well with our goal. Boundary loss complements Dice and is slightly edge-sensitive, focusing on mask contours which is relevant in ellipse fitting. The exact expressions for the Dice, Focal, and Boundary losses can be found in Appendix A.

4.3 Implementation

As a considerable number of mathematical operations are executed during the learning process of a CNN model, a standard central processing unit (CPU) would take a significant amount of time to perform them. For that, all of our calculations were run in an environment with CUDA installed, to enable us the use of a graphics processing unit (GPU) to speed up those intensive computational tasks by incorporating parallel computing. All code is implemented in Python, using a custom Spyder environment configuration and importing a selection of packages, including Tensorflow, Keras and Astropy for FITS file handling and coordinate transformations.

See Appendix B for more information on the environment and the equipment used.

5 Experimental Results & Discussion

5.1 Binary Classification Performance

To evaluate the classification capability of our ResUNet models trained on Spitzer 8 μm data and masks provided from MWP, we first assess their ability to distinguish whether an image contains an interstellar bubble or not. We experiment with three different loss functions during training: Dice loss, Focal loss, and Boundary loss. See Appendix A for more information on the environment and the equipment used.Their performance is summarized in Table 1 and Table 2.

TABLE 1: Confusion matrices for image-level classification under different loss functions.

Dice Loss

	Pred. Neg.	Pred. Pos.
Actual Neg.	1057	108
Actual Pos.	32	273

Focal Loss

	Pred. Neg.	Pred. Pos.
Actual Neg.	975	190
Actual Pos.	26	279

Boundary Loss

	Pred. Neg.	Pred. Pos.
Actual Neg.	0	1165
Actual Pos.	0	305

TABLE 2: Evaluation metrics for each loss function on the image-level classification task.

Metric	**Dice Loss**	**Focal Loss**	**Boundary Loss**
Accuracy	0.905	0.881	0.208
Precision	0.717	0.595	0.208
Recall	0.895	0.915	1.000
F1-score	0.796	0.721	0.344
ROC-AUC	0.901	0.894	0.500

From the tables above, it is evident that Dice loss achieves the best overall balance between precision and recall, resulting in the highest F1-score among the three. While Focal loss obtains a slightly higher recall (0.915), there is an expense in precision, indicating a higher number of false positives. Boundary loss, on the other hand, fails to provide meaningful classification results, simply by predicting all inputs as positive and achieving an artificially high recall of 1.0 with poor precision.

These results demonstrate that Dice loss is the most suitable choice for our task, effectively handling the inherent class imbalance in the data and maintaining a balance between sensitivity and specificity.

5.2 Training Performance and Loss Curves: Dice vs. Focal Loss

Given the strong classification results from both Dice and Focal loss, we next compare their training behavior over several epochs. Figure 5 shows the training and validation loss and accuracy curves for the two models.

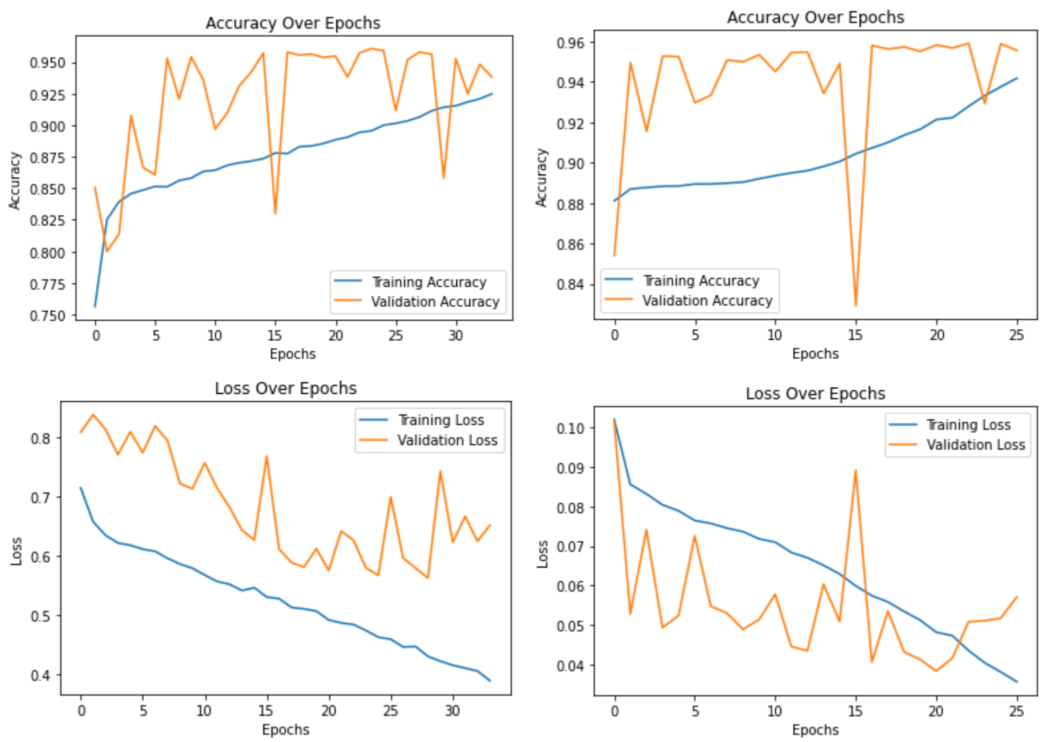

FIGURE 5: Training and validation performance across epochs for the two best-performing loss functions: Dice Loss (left) and Focal Loss (right). Top row shows accuracy curves; bottom row shows corresponding loss curves.

For Dice loss, the training and validation losses show a consistent downward trend, with the model continuing to learn steadily across 25 epochs. Validation accuracy stabilizes above 0.95, while training accuracy increases more slowly, suggesting good generalization and limited overfitting.

In contrast, the Focal loss model displays more erratic validation behavior. Training and validation losses quickly plateau, and although the validation accuracy remains high, the training accuracy plateaus significantly lower. This gap may indicate underfitting, where the model struggles to learn from the more complex weighting dynamics introduced by Focal loss.

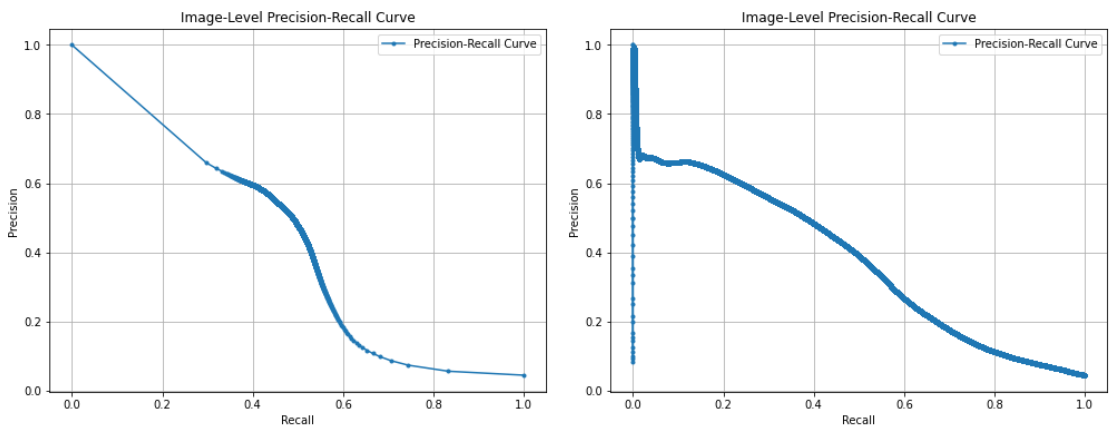

FIGURE 6: Image-level precision–recall curves for the Dice Loss (right) and Focal Loss (left) predictions.

Precision-recall analysis in Figures 6 supports this interpretation. The curve for Dice loss demonstrates a more balanced trade-off between precision and recall. Focal loss tends to favor higher recall at the expense of precision, which may explain its weaker precision score in the evaluation metrics.

Taken together, these findings confirm the selection of Dice loss as the most suitable for our binary classification task, providing both stable training and optimal performance under class imbalance conditions.

5.3 Post-Processing & Segmentation

To extract useful physical properties from the binary masks produced by our model, we implement a post-processing pipeline. The goal is to identify distinct bubble structures and characterize them geometrically. The procedure consists of three main stages: connected component labeling, shape filtering, and ellipse fitting.

First, predicted binary masks are smoothed using a Gaussian filter and cleaned with morphological closing operations to remove small, false foreground regions. The cleaned mask is then passed through a connected component labeling algorithm, which assigns unique identifiers to each contiguous region of non-zero pixels. Components with fewer than 30 pixels or an irregular shape are discarded, as this signifies the existence of irrelevant traces.

For the remaining regions, we fit ellipses using the OpenCV implementation of the `fitEllipse` method. Each bubble is then assigned the following:

- a centroid in pixel coordinates - converted to sky coordinates via WCS
- a semi-major and a semi-minor axis length - to compute the effective radius in arcminutes
- an eccentricity
- and a position angle.

This geometric approximation provided a compact and physically interpretable representation of each detected structure. See Figure 7 for visual examples of predicted contours and fitted ellipses.

To further analyze the distribution of our model's detected bubbles, we compare the effective radii of the predicted masks against those listed in the MWP catalog. Figure 8. shows that our model identifies a significantly larger number of smaller bubbles. This behavior is expected, due to the fixed 256×256 input window used during training, which is better suited for detecting compact or partially visible bubble-like structures. Many of

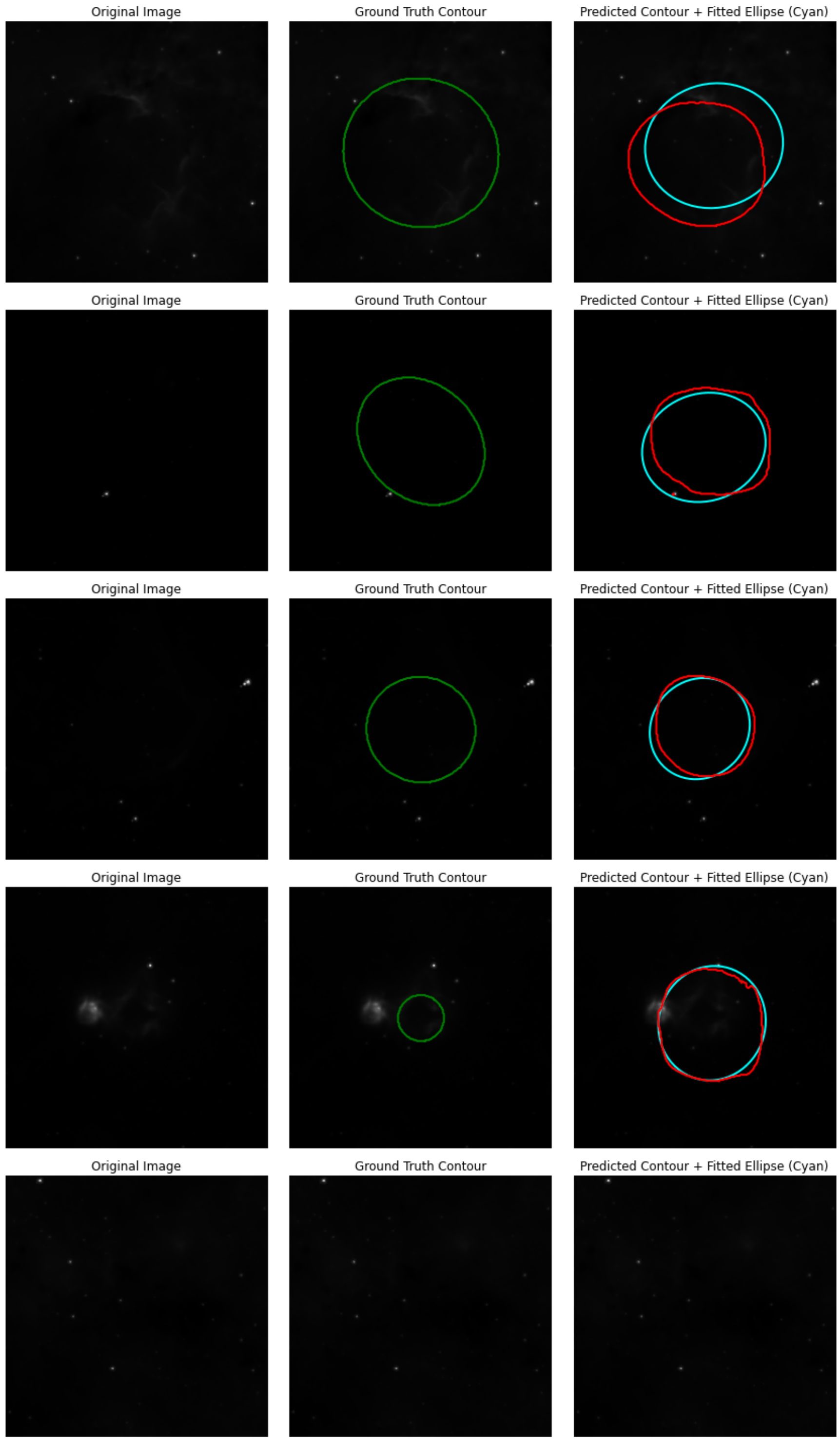

FIGURE 7: Examples of bubble predictions with fitted ellipses (cyan) overlaid on model outputs. Ground truth contours are shown in green and predicted contours in red.

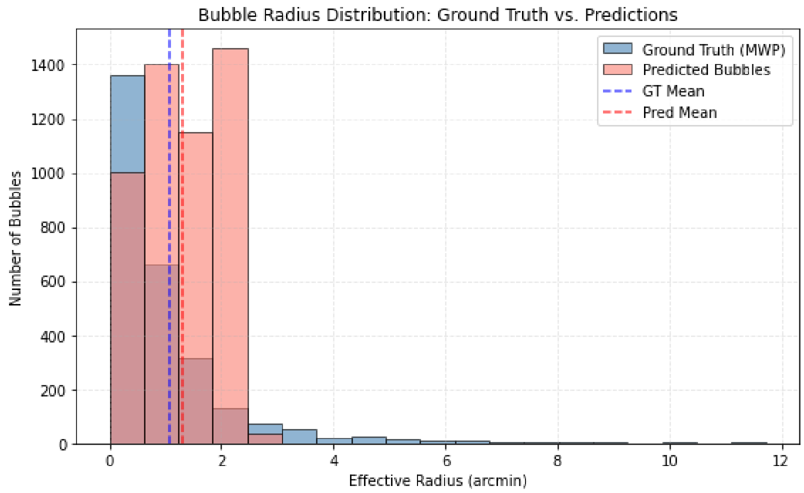

FIGURE 8: Histogram comparing effective bubble radii from the MWP catalog and predicted masks. Dashed lines indicate mean values for each distribution.

these smaller detections may correspond to incomplete rims, faint shells, or other fragments which were overlooked in the original visual classification. This also suggests that our CNN model may be more sensitive to subtle morphological features than manual methods, such as the texture of PAH rims and shape symmetry. This sensitivity allows the model to generalize more effectively and recognize bubble-like patterns beyond the specific examples it encountered during training.

A sample of the extracted bubbles from the test set is provided in Table 3.

TABLE 3: Sample of Predicted Bubbles from Post-Processing

RA (deg)	Dec (deg)	Radius (arcmin)	Eccentricity	Position Angle (°)
31.779	-0.264	1.83	0.82	0.06
33.486	-0.008	0.27	0.69	0.09
33.614	0.034	1.89	0.40	148.80
37.695	0.034	1.99	0.31	155.10
43.254	-0.008	2.19	0.43	10.94
...				

This table visualizes the diversity of detected structures in terms of both morphology and spatial coverage. While additional filtering could refine the catalog, the raw predictions provide a promising foundation for locating existing bubble structures in this dataset.

5.4 Sky Region Comparison

To qualitatively assess the performance of the model in a high-density environment, after careful consideration, we select the region of the Galactic plane that spans from longitudes 19.5° to 28.5° for visualization and analysis. This range was chosen because it overlaps with the intersection of the Scutum–Centaurus and Sagittarius spiral arms—a well-known star-forming complex rich in infrared features and a high density of MWP bubbles. This makes it an ideal test for evaluating both model accuracy and potential over-segmentation or under-segmentation in cluttered environments.

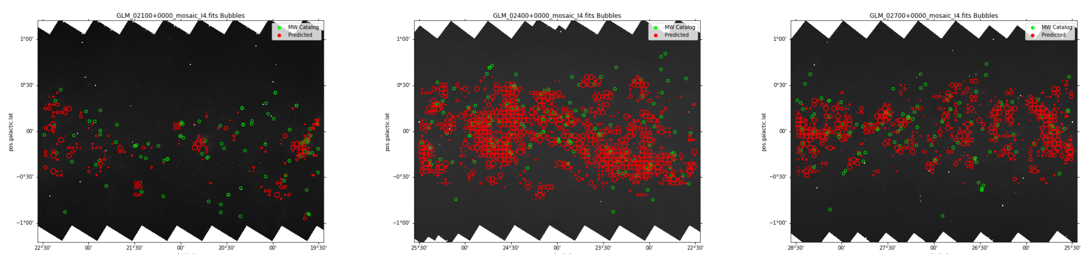

FIGURE 9: Spatial distribution of predicted (red) and catalog (green) bubbles across selected GLIMPSE fields.

Figure 9 displays sample GLIMPSE mosaic images from this region with bubble overlays. Green ellipses indicate annotated bubbles from the Milky Way Project (MWP DR1), while red ellipses represent the model's predicted bubble candidates after post-processing. The overlay uses WCS coordinate transforms to ensure consistent alignment between catalogs.

Visual inspection suggests that the model successfully recovers the majority of cataloged bubbles, particularly larger and well-defined structures. In addition, the model detects several smaller or partially overlapping shells not included in the MWP annotations. These often appear along filamentary PAH arcs or at the edges of brighter emission regions, suggesting that the CNN may have learned to associate subtle spatial textures with bubble morphology.

In crowded areas there are some discrepancies, as the model occasionally fragments a large MWP bubble into multiple smaller detections. This is consistent with its fixed input size and could be improved through hierarchical or multi-scale input designs. Nonetheless, the predictions align well with the known structure of the interstellar medium in this region, reinforcing the model's ability to generalize to complex sky environments.

5.5 Comparison with other Models

Recent efforts to automate bubble detection in Galactic surveys include both traditional machine learning and deep learning approaches. Among these, BRUT from Beaumont et al., 2014 and CASI from Van Oort et al., 2019 represent two notable models with different goals and methodologies.

BRUT relied on manually extracted features from Spitzer images, such as ring-like brightness profiles and edge sharpness. These features were input to a Random Forest classifier to assign bubble likelihood scores to regions of interest. While BRUT achieved moderate success in identifying clear bubble candidates, its reliance on manually defined heuristics limited its ability to generalize across complex or faint morphologies.

CASI, on the other hand, utilized convolutional neural networks trained on MWP data to classify cutouts as "bubble" or "non-bubble." However, CASI focused strictly on binary classification without providing any localization or segmentation of bubble boundaries, which restricted its utility for morphological analysis.

In contrast, this study offers a full segmentation pipeline implementation, one trained solely on real elliptical masks derived from the MWP catalog. This allows not only the identification of bubble-containing regions, but also the precise extraction of geometric parameters such as effective radius, eccentricity, and position angle—enabling quantitative comparison between predicted and cataloged structures.

By combining DL with shape-fitting post-processing, this work bridges the gap between classification and catalog-level morphological inference, offering a scalable and interpretable framework for Galactic bubble studies.

6 Conclusions & Future Work

This study presents a Convolutional Neural Network approach for segmenting interstellar bubbles in the Spitzer GLIMPSE 8 μm survey. Trained directly on masks derived from the Milky Way Project (MWP), our model performs pixel-wise segmentation of potential bubble regions and enables the extraction of morphological parameters through post-processing ellipse fitting.

Evaluation against MWP-labeled regions demonstrates that the model is capable of capturing both well-defined and subtle bubble features. Despite yielding promising results, this study has several limitations that define its current scope and suggest directions for future improvement:

- Limited training set size: The model is trained exclusively on a relatively small dataset of cutouts derived from MWP DR1 bubbles, constraining the diversity of bubble morphologies and environments it is exposed to during learning.
- Single-band input: Only the 8 μm Spitzer band is used as input. While this wavelength indicates PAH emission and bubble rims, it excludes key complementary features visible in other bands, potentially leading to misclassification of regions lacking strong 8 μm contrast.
- Texture bias and false positives: The CNN occasionally identifies PAH filaments, yellow balls, bow shocks, or circular noise patterns as bubble candidates because of Deep Learning's known texture bias. These false positives may not correspond to physically coherent shell structures.
- No post-prediction physical filtering: The model does not apply astrophysical constraints, which can result in implausible or poorly defined detections that would typically be filtered out in a crowd-sourced catalog.

The points above indicate opportunities for refinement. Future work should aim to address these by:

- Expanding the training dataset with multi-band and multi-survey data to improve generalization.
- Incorporating multi-scale segmentation techniques to better capture large or overlapping structures.
- Applying astrophysical reasoning to filter implausible predictions.
- Deploying the model on other infrared surveys (e.g., WISE, JWST MIRI) to explore transferability across instruments.

By combining CNN-based segmentation with geometric analysis, this framework contributes a robust and extensible tool for advancing bubble morphology studies in star formation.

References

Beaumont, C. N., Goodman, A. A., Kendrew, S., Williams, J. P., & Simpson, R. (2014). The milky way project: Leveraging citizen science and machine learning to detect interstellar bubbles. *The Astrophysical Journal Supplement Series*, *214*(1), 3. https://doi.org/10.1088/0067-0049/214/1/3

Bolatto, A. D., Wolfire, M., & Leroy, A. K. (2013). The co-to-h2 conversion factor. *Annual Review of Astronomy and Astrophysics*, *51*, 207–268. https://doi.org/10.1146/annurev-astro-082812-140944

Churchwell, E., Povich, M. S., Allen, D., Taylor, M. G., Meade, M. R., Babler, B. L., Indebetouw, R., Watson, C., Whitney, B. A., & Wolfire, M. G. (2006). The bubbling galactic disk. *The Astrophysical Journal*, *649*(2), 759–778. https://doi.org/10.1086/506185

Deharveng, L., Schuller, F., Anderson, L. D., Zavagno, A., Wyrowski, F., Menten, K. M., Bronfman, L., Testi, L., & Walmsley, C. M. (2010). A gallery of bubbles: The nature of the bubbles observed by spitzer. *Astronomy & Astrophysics*, *523*, A6. https://doi.org/10.1051/0004-6361/201014652

Dieleman, S., Willett, K. W., & Dambre, J. (2015). Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly Notices of the Royal Astronomical Society*, *450*(2), 1441–1459. https://doi.org/10.1093/mnras/stv632

Domínguez Sánchez, H., Huertas-Company, M., Bernardi, M., Tuccillo, D., & Fischer, J. L. (2020). Deep learning for galaxy morphology classification. *Monthly Notices of the Royal Astronomical Society*, *499*(1), 2617–2635. https://doi.org/10.1093/mnras/staa2953

Elmegreen, B. G., & Lada, C. J. (1977). Sequential formation of subgroups in OB associations., *214*, 725–741. https://doi.org/10.1086/155302

Geen, S., Rosdahl, J., Blaizot, J., Devriendt, J., & Slyz, A. (2021). The geometry and dynamical role of stellar wind bubbles in photoionized h ii regions. *Monthly Notices* *of the Royal Astronomical Society*, *501*(1), 1352–1367. https://doi.org/10.1093/mnras/staa3645

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., & Brendel, W. (2022). Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. https://arxiv.org/abs/1811.12231

Islam, M. A., Kowal, M., Esser, P., Jia, S., Ommer, B., Derpanis, K. G., & Bruce, N. (2021). Shape or texture: Understanding discriminative features in cnns. *International Conference on Learning Representations*. https://arxiv.org/abs/2101.11604

Kendrew, S., Simpson, R., Bressert, E., Povich, M. S., Robitaille, T. P., Schmeja, S., Arvidsson, K., Berriman, G. B., Boulanger, F., Brunt, C. M., Calzetti, D., Carey, S. J., Churchwell, E., Cohen, M., Cyganowski, C. J., Indebetouw, R., Jackson, J. M., Kobulnicky, H. A., Longmore, S. N., . . . Yusef-Zadeh, F. (2012). The milky way project: A statistical study of massive star formation associated with infrared bubbles. *The Astrophysical Journal*, *755*(1), 71. https://doi.org/10.1088/0004-637X/755/1/71

Kim, J., Lee, J., Chevance, M., & et al. (2023). Connecting stellar and galactic scales: Energetic feedback from stellar wind bubbles to supernova remnants. *Monthly Notices of the Royal Astronomical Society*, *526*(3), 4054–4072. https://doi.org/10.1093/mnras/stad2864

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Nishimoto, S., Onishi, T., Nishimura, A., Fujita, S., Kawanishi, Y., Nakatani, S., Tokuda, K., Shimajiri, Y., Kaneko, H., Miyamoto, Y., Inoue, T., & Ito, A. M. (2025). Infrared bubble recognition in the milky way and beyond using deep learning. *Publications of the Astronomical Society of Japan*, *77*(2), 403–424. https://doi.org/10.1093/pasj/psaf008

Peeters, E., Spoon, H. W. W., & Tielens, A. G. G. M. (2004). Polycyclic aromatic hydrocarbons as a tracer of star formation? *The Astrophysical Journal*, *613*(2), 986–1003. https://doi.org/10.1086/423237

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. https://arxiv.org/abs/1505.04597

Salama, F. Pahs in astronomy: A review. In: *Organic matter in space, proceedings iau symposium no. 251*. 2008, 357–366. https://doi.org/10.1017/S1743921308021915.

Schawinski, K., Zhang, C., Zhang, H., Fowler, L., & Santhanam, G. (2017). Generative adversarial networks recover features in astrophysical images of galaxies beyond the deconvolution limit. *Monthly Notices of the Royal Astronomical Society: Letters*, *467*(1), L110–L114. https://doi.org/10.1093/mnrasl/slx008

Simpson, R. J., Povich, M. S., Kendrew, S., Lintott, C. J., Bressert, E., Arvidsson, K., Cyganowski, C., Maddison, S. T., Schawinski, K., Sherpa, D., & et al. (2012). The milky way project first data release: A bubblier galactic disc. *Monthly Notices of the Royal Astronomical Society*, *424*(4), 2442–2460. https://doi.org/10.1111/j.1365-2966.2012.21483.x

Smith, M. J., & Geach, J. E. (2023). Astronomia ex machina: A history, primer and outlook on neural networks in astronomy. *Royal Society Open Science*, *10*(5). https://doi.org/10.1098/rsos.221454

Tielens, A. G. G. M. (2005). *The physics and chemistry of the interstellar medium*. Cambridge University Press.

Van Oort, B., Lee, M.-Y., Peek, J. E. G., & Chang, P. (2019). Casi: A convolutional neural network approach for shell identification. *The Astrophysical Journal*, *880*(1), 83. https://doi.org/10.3847/1538-4357/ab2a7d

Visser, R., van Dishoeck, E. F., & Black, J. H. (2009). The photodissociation and chemistry of co isotopologues: Applications to interstellar clouds and circumstellar disks.

Astronomy & Astrophysics, *503*(2), 323–343. https://doi.org/10.1051/0004-6361/200912408

Vojtekova, A., Lieu, M., Valtchanov, I., Altieri, B., Old, L., & et al. (2021). Learning to denoise astronomical images with u-nets. *Monthly Notices of the Royal Astronomical Society*, *503*(3), 3204–3215. https://doi.org/10.1093/mnras/stab663

Watkins, E. J., Barnes, A. T., Henny, K., Kim, H., Kreckel, K., Meidt, S. E., Klessen, R. S., Glover, S. C. O., Williams, T. G., Keller, B. W., Leroy, A. K., Rosolowsky, E., Lee, J. C., Anand, G. S., Belfiore, F., Bigiel, F., Blanc, G. A., Boquien, M., Cao, Y., ... Whitmore, B. C. (2023). Phangs–jwst first results: A statistical view on bubble evolution in ngc 628. *The Astrophysical Journal Letters*, *944*(2), L24. https://doi.org/10.3847/2041-8213/aca6e4

Xu, D., Offner, S. S. R., Gutermuth, R., & Van Oort, C. (2020a). Application of convolutional neural networks to identify protostellar outflows in co emission. *The Astrophysical Journal*, *905*(2), 172. https://doi.org/10.3847/1538-4357/abc8d2

Xu, D., Offner, S. S. R., Gutermuth, R., & Van Oort, C. (2020b). Application of convolutional neural networks to identify stellar feedback bubbles in co emission. *The Astrophysical Journal*, *890*(1), 64. https://doi.org/10.3847/1538-4357/ab65f5

Appendix A: Loss Functions

Boundary Loss

Boundary Loss:

$$\mathcal{L}_{\text{Boundary}} = \int_{\Omega} \phi_G(x) \cdot p(x)\, dx$$

where:

- $\phi_G(x)$ = distance-to-boundary map of the ground truth mask G at location x
- $p(x)$ = predicted probability at location x
- Ω = image domain

Focal Loss

Focal Loss:

$$\mathcal{L}_{\text{Focal}} = -\sum_i \alpha(1 - p_i)^{\gamma} g_i \log(p_i) + (1 - \alpha) p_i^{\gamma} (1 - g_i) \log(1 - p_i)$$

where:

- p_i = predicted probability for pixel i
- g_i = ground truth label for pixel i
- $\alpha \in [0, 1]$ = class balancing factor
- $\gamma \geq 0$ = focusing parameter (~ 2)

Dice Loss

Dice Coefficient (DSC):

$$\text{DSC} = \frac{2 \cdot |A \cap B|}{|A| + |B|}$$

where:

- A = predicted binary mask
- B = ground truth binary mask

Dice Loss:

To use it as a loss function (i.e., to minimize it), we subtract from 1:

$$\mathcal{L}_{\text{Dice}} = 1 - \frac{2 \sum_i p_i g_i + \epsilon}{\sum_i p_i + \sum_i g_i + \epsilon}$$

where:

- p_i = predicted probability for pixel i
- g_i = ground truth label for pixel i
- ϵ = small constant to avoid division by zero (we used 10^{-6})

Appendix B: Equipment details

The hardware specifications and software used in this dissertation are listed as follows:

- Hardware specifications:

  CPU: AMD Ryzen 5 5600X 6-Core Processor, 3.7 GHz, 6 Cores

  GPU: NVIDIA GeForce RTX 3060

  RAM: 32 GB DDR4

  Storage: 1 TB

- Software environment:

  Operating system: Windows 11

  NVIDIA driver version:

  CUDA version: 11.8.89

  Python version: 3.10.8

- Main Python libraries:

  Astropy: Version: 6.1.7

  Matplotlib: 3.6.3

  NumPy: 1.25.0

  Pandas: 2.2.3

  TensorFlow: 2.10.1

  Keras: 2.10.0

  Scipy: 1.15.1

For the rest of the Python libraries, please refer to `requirements.txt`[4]

[4]Available at https://github.com/Bravebirdie/Detecting-Interstellar-Bubbles-in-Images-using-CNNs