Fall 11-15-2016

Automatic Segmentation of Coronal Holes in Solar Images and Solar Prediction Map Classification

Venkatesh Jatla
University of New Mexico

Venkatesh Jatla

Candidate

Electronics and Computer Engineering

Department

This thesis is approved, and it is acceptable in quality and form for publication:

Approved by the Thesis Committee:

Prof. Marios Pattichis
, Chairperson

Prof. Mark Gilmore

Dr. Nick Charles Arge

Automatic Segmentation of Coronal Holes in Solar Images and Solar Prediction Map Classification

by

Venkatesh Jatla

B.Tech., Electronics and Communications, Vellore institute of Technology, 2007

THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science
Computer Engineering

The University of New Mexico

Albuquerque, New Mexico

December, 2016

Dedication

This thesis is dedicated to the Sun, by whose grace earth shines and whose warmth has always given me joy.

Acknowledgments

Automatic Segmentation of Coronal Holes in Solar Images and Solar Prediction Map Classification

by

Venkatesh Jatla

B.Tech., Electronics and Communications, Vellore institute of Technology, 2007

M.Sc., Computer Engineering, University of New Mexico, 2016

Abstract

Solar image analysis relies on the detection of coronal holes for predicting disruptions to earth's magnetic field. The coronal holes act as sources of solar wind that can reach the earth. Thus, coronal holes are used in physical models for predicting the evolution of solar wind and its potential for interfering with the earth's magnetic field. Due to inherent uncertainties in the physical models, there is a need for a classification system that can be used to select the physical models that best match the observed coronal holes.

The physical model classification problem is decomposed into three subproblems. First, the thesis develops a method for coronal hole segmentation. Second, the thesis develops methods for matching coronal holes from different maps. Third, based on the matching results, the thesis develops a physical map classification system.

A level-set segmentation method is used for detecting coronal holes that are observed in extreme ultra-violet images (EUVI) and magnetic field images. For

v

validating the segmentation approach, two independent manual segmentations were combined to produce 46 consensus maps. Overall, the level-set segmentation approach produces significant improvements over current approaches.

Coronal hole matching is broken into two steps. First, an automated method is used to combine coronal holes into clusters. Second, a Linear Program formulation is used for matching the clusters. The results are validated using manual clustering and matching. Compared to manual matching, the automated matching method gave more than 85 percent accuracy.

Physical map classification is based on coronal hole matching between the physical maps and (i) the consensus maps (semi-automated), or (ii) the segmented maps (fully-automated). Based on the matching results, the system uses area differences, shortest distances between matched clusters, number and areas of new and missing coronal hole clusters to classify each map. The results indicate that the automated segmentation and classification system performs better than individual humans.

Contents

Contents

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Coronal holes represent regions of the solar surface where the plasma density is low. Inside the coronal holes, the magnetic field lines are open and stretch into space. Thus, coronal holes can be sources of solar winds that can escape at high speeds. Depending on its location, a solar wind that originates from a coronal hole can reach the earth and cause large disturbances to the earth's magnetic field. If the disturbances are sufficiently large, they can cause significant damage to the power and communications infrastructure. To avoid such damage, it is important to develop reliable methods for predicting the locations and the sizes of the coronal holes.

Coronal holes can be observed in solar images taken in various spectral bands. Based on the detected coronal holes, physical models can then be used to predict solar wind propagation. Unfortunately, there are strong variations in the physical models that can be used for forecasting. There is thus a strong need to reliably detect the coronal holes and select physical models that match the detected coronal holes to support accurate solar predictions. A fundamental problem associated with developing analysis methods comes from the lack of reliable maps that outline the locations of the coronal holes.

(a) EUVI image (synoptic image).

(b) Magnetic image (photomap image).

(c) Consensus map.

(d) Current method: `unit_dist` $= 0.17$
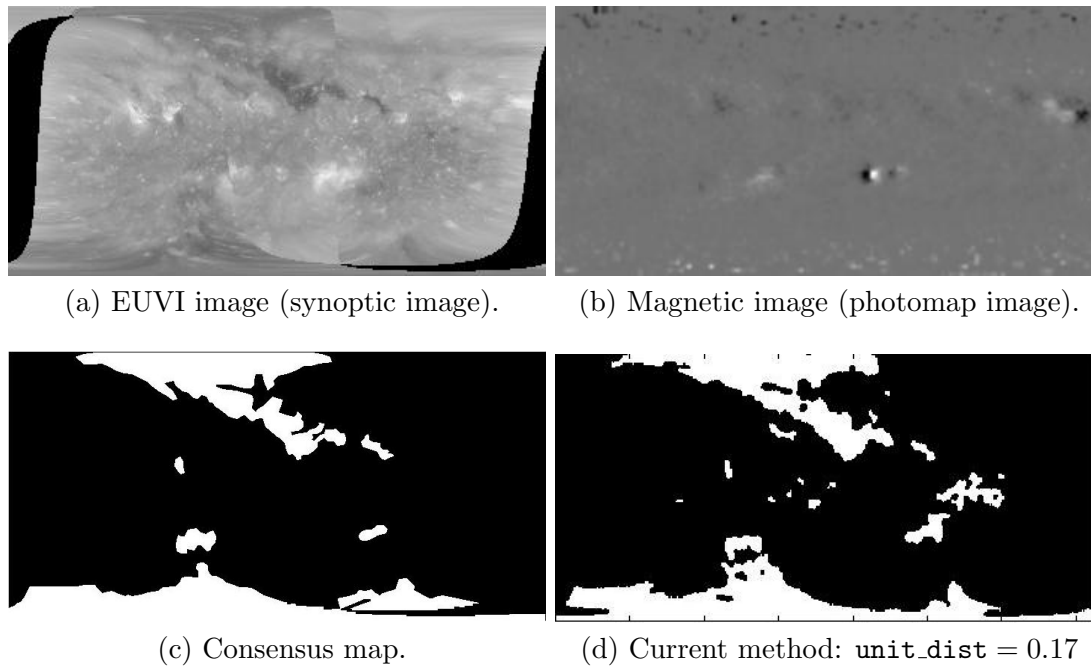
Figure 1.1: The basic coronal hole detection problem. Example input data maps for May 2, 2011 are shown: EUVI (a) and magnetic image in (b). The consensus map for the coronal holes is shown in (c). Results from using the current method described in [22] can be found in (d). The `unit_dist` refers to the distance of the current algorithm from the ideal performance given by (`sensitivity`, `specificity`) $= (1, 1)$, where the ideal image is given in (c).

To assess the performance of methods for detecting coronal holes, there is a need to establish a ground truth from solar observations. Unfortunately, the standard practice of comparing automatic and manual annotations from a single expert is not acceptable since such an approach can lead to significant biases. A manual annotation tool developed by Dr. Pattichis in the Summer of 2012 has been used to allow multiple users to manually outline the locations and regions of coronal holes within specific dates. The system was used to generate coronal hole maps by two different users, and then again to derive consensus annotations by combining the two

independent maps. The consensus maps have the unique advantage that they avoid biases associated with individual raters.

This thesis develops automated image processing models that can be used to detect coronal holes automatically (see Fig. 1.1) and then classify physical models on whether they correctly match the coronal holes that have been detected. The basic approach relies on the detection of coronal holes based on extreme ultra-violet images (EUVI) and magnetic images. An automated segmentation method has been developed that improves significantly over a previously considered method described in [22]. The thesis also extends prior, initial work by Andrew Delgado to address the matching problem.

To develop the necessary methods, the thesis developed and implemented manual classification protocols that were used to establish a ground truth database for the classification of physical models. The manual classification is used to identify the best physical models based on how well they match the consensus maps.

We present an example of the physical model classification problem in Figs. 1.2 and 1.3. The basic idea is to develop a classification system that grades physical models based on their agreement with the consensus map. The physical maps that are closer to the consensus map are classified as good for use in forecasting. On the other hand, the maps that are significantly different from the consensus map, are to be classified as unsuitable for use for forecasting.

1.1   Thesis Statement

My thesis is that I can develop an automated system that can be used to classify physical models based on how well they can be used to predict the location and areas of coronal holes that are themselves detected from solar images.

1.2 Contributions

The primary contribution of the thesis include:

- ***Manual classification of physical models:*** The thesis provides a protocol for the manual classification of physical models based on how well they match coronal holes seen in consensus maps.

- ***Level sets segmentation of coronal holes:*** The thesis describes a hybrid method that uses magnetic images and EUVI to detect coronal holes.

- ***Coronal holes matching algorithm:*** The thesis describes a matching method that is used for matching clusters of coronal holes between maps and also to detect new (generated) and missing (removed) coronal holes.

- ***Computer classification of physical maps:*** An automated algorithm has been developed for selecting physical maps that should be used for solar prediction. The computer classification methods has been validated using manual classification.

1.3 Thesis Overview

The remainder of the thesis is organized into 5 chapters:

- **Chapter 2: Background.** This chapter describes prior work.

- **Chapter 3: Manual classification of physical models.** This chapters describes the visual interface and the protocol that was used to create a ground truth of manual classifications for the physical models.

- **Chapter 4: Computer segmentation and classification of physical models.** This chapter describes the coronal hole segmentation, matching algorithm and the automated classification system that is used to select the best physical models.

- **Chapter 5: Results.** This chapter provides a summary of the results for coronal hole segmentation, matching and computer classification, as compared to manual classification.

- **Chapter 6: Conclusion and future work.** This chapter provides a summary of the thesis and recommendations for future work.

(a) Consensus map

(b) Model 1

(c) Model 2

(d) Model 3

(e) Model 4

(f) Model 5

(g) Model 6

Figure 1.2: Physical model classification problem (I of II). Physical models are classified based on their distance from the consensus map. The maps correspond to 21-01-2011.

(a) Consensus map

(b) Model 7

(c) Model 8

(d) Model 9

(e) Model 10

(f) Model 11

(g) Model 12

Figure 1.3: Physical model classification problem (II of II). Physical models are classified based on their distance from the consensus map. The maps correspond to 21-01-2011.

Chapter 2

Background

2.1 Creating consensus maps

Groud truth is necessary in order to assess the performace of coronal hole detection methods. Unfortunately, the standard practice of comparing automatic and manual annotations from a single expert is not acceptable since such an approach can lead to significant biases. A manual annotation tool developed by Dr. Pattichis in the Summer of 2012 has been used to allow multiple users to manually outline the locations and regions of coronal holes within specific dates. The system was used to generate coronal hole maps by two different users, and then again to derive consensus annotations by combining the two independent maps. The consensus maps have the unique advantage that they avoid biases associated with individual raters. Two Carrington rotations are segmented giving 50 consensus maps. Fig. 2.1 shows the IDL framework developed to segment coronal holes. .
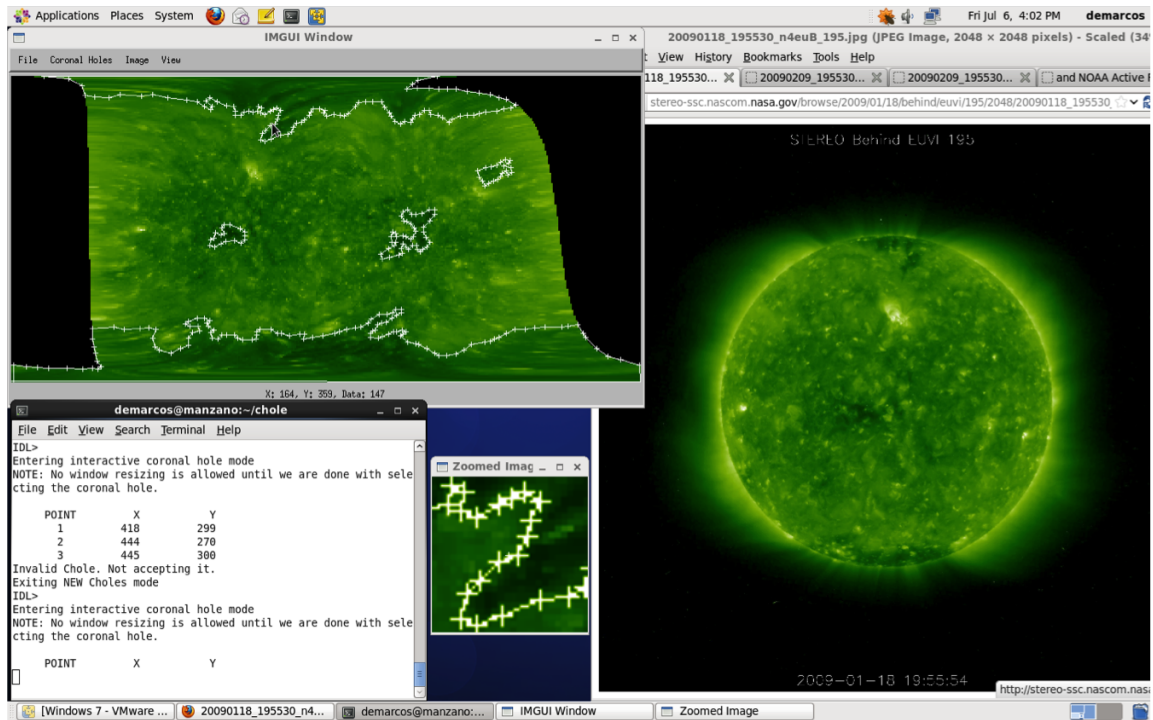
Figure 2.1: Manual annotation framework developed to segment coronal holes.

2.2   Coronal hole segmentation

Coronal holes [5] have properties detectable using EUVI and photo maps. They appear darker in EUV (Extreme Ultra Violet) spectrum, unipolar, and do not cross magnetic neutral lines [6]. Based on coronal hole properties methods like [22, 26] are developed to segment coronal holes automatically. In Section 2.2.1 we describe Henney Harvey method [22], an earlier method that provides a careful integraiton of unipolarity and darker region requirements.

2.2.1 Henney Harvey method

The algorithm takes the EUVI and magnetic (photomap) images as inputs and returns segmented coronal hole image. The input images are resized to ensure that they share the same resolution.

An initial segmentation is estimated on the EUVI image. First, the dark regions in the EUVI images are detected using spatially adaptive thresholding (see Fig. 2.2). The need for adaptive thresholding comes from the fact that the EUVI images map the spherical surface of the sun to a rectangular image with pixels of variable size. These dark regions are denoised using a morphological `open-close`, followed by an area open that removes small blobs.

Magnetic constraints are then imposed on each estimated EUVI component based on the corresponding regions in the photomap (magnetic) image. To this end, the algorithm needs to determine the polarity of each EUVI component. An averaging filter is first applied to the polarity image to reduce the noise level. The polarity of each image is then computed as a ratio of the dominant polarity to the total number of pixels. The unipolar assumption is enforced by removing blobs for which the polarity ratio does not exceed 65%. The remaining blobs form the estimated coronal hole image.

2.3 Selecting solar models by matching coronal holes

The thesis extends prior, initial work by Andrew Delgado where he defined and solved the problem of automatically selecting physical models based on observations. The best physical models are selected by matching the coronal holes between model and consensus. For each one of the 12 physical models, an optimal physical model is

function HENNEYHARVEY(EUVI, `mag`)

▷ Threshold with area correction.
▷ θ comes from spherical coordinates.
▷ μ, σ are estimated over local windows.
$T \leftarrow \mu - \sigma(0.7 + 0.1\cos(\theta))$
`init_dark_img` $\leftarrow$ (EUVI $< T$)

▷ Reduce noise in EUVI image
`den_img` $\leftarrow$ **open-close** `init_dark_img` with SE
`den_img2` $\leftarrow$ **remove** small blobs from `den_img`
              with `area`< 25

▷ Denoise and prepare polarity image
`blurred`$\leftarrow$**blur** `mag`.
`pol_img`$\leftarrow$**compute_blob_polarity** of `blurred`

▷ Keep unipolar blobs only
`coronal_hole_img` $\leftarrow$ **remove** non-unipolar
                blobs from `den_img2`
              with $< 65\%$ in `pol_img`

`r`eturn `coronal_hole_img`
end function

Figure 2.2: Henney-Harvey method for detecting coronal holes.

manually selected for validating the approach. Over 50 maps, the results indicated that there is a 52% agreement between the maps that are visually selected and the ones selected by the automated approach. A 60% accuracy is acheived on removing 9 maps for which visual matching was of low-confidence.

The report does not work with coronal hole polarity. This might have resulted in incorrect clustering and matching of coronal holes having different polarities. In addition, no clear protocol was followed when manually selecting the best model. This thesis addressess these issues by carefully working with polarity, and following protocol for manually selecting the best model.

Chapter 3

Manual classification of physical models

In this chapter, we develop a system to support manual classification of physical models. The problem is introduced in section 3.1. Section 3.2 describes the manual protocol.

3.1   Introduction

The basic problem of classifying physical models is depicted in Fig. 3.1. The basic idea is to compare physical model against the consensus maps and then select the physical models that are closest to the consensus map. Physically, coronal hole matching requires that we compare clusters of coronal holes that share the same polarity, similar location, and similar area. To appreciate the problem, we note that we have:

1. **Matching of many to many:** To see the problem, note that the consensus

(a) Consensus image, size=720 × 360   (b) Model 1 prediction image, size=144 × 72

(c) Model 3 prediction image, size=144 × 72   (d) Model 11 prediction image, size=144 × 72
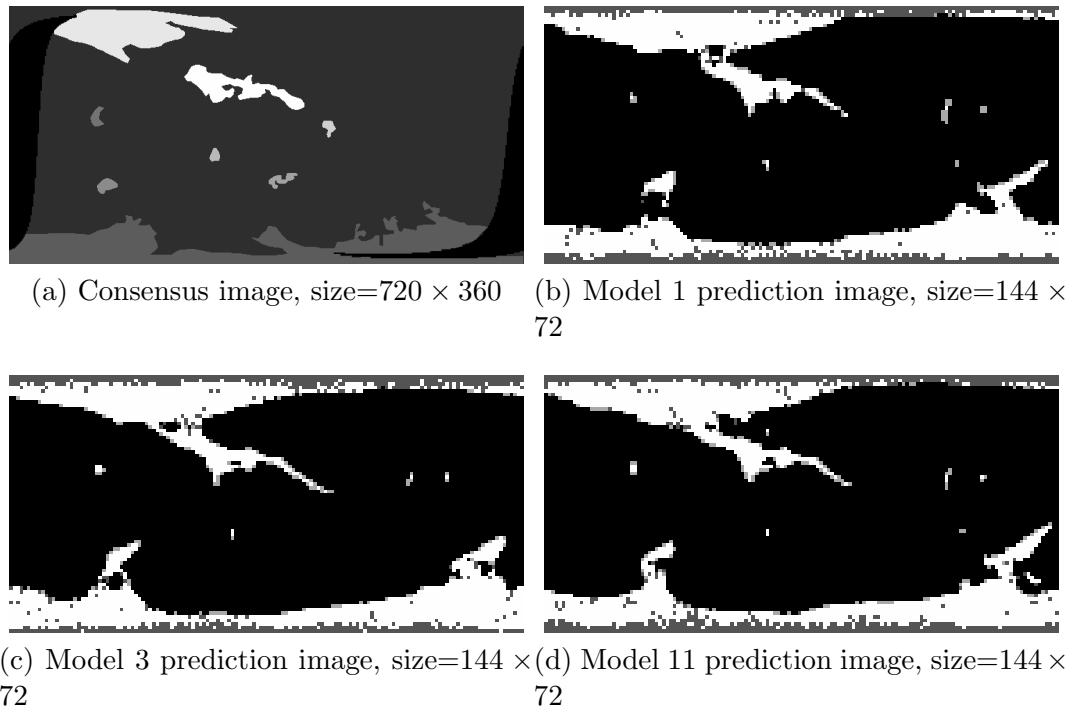
Figure 3.1: The physical model classification problem.

map of Fig. 3.1 has 8 coronal holes, compared to 6 for model 3 (see Fig. 3.1c), and 11 for model 11 (see 3.1d).

2. **Classification consistency:** Many physical models appear similar. Thus, physical models need to be grouped together based on their similarities. Then, instead of classifying individual physical maps, we need to classify entire groups.

3. **Missing observation regions:** The darkest regions of the consensus maps represent regions of no observations. Matching needs to take into account that we cannot match coronal holes in these regions.

4. **Projection effects** The areas of polar coronal holes are greatly exaggerated due to projection effects. To avoid such issues, we will not consider visible (manual) matching of regions that are below 30 and above 150 degrees.

| (a) Annotated consensus image. | (b) Physical model image. |
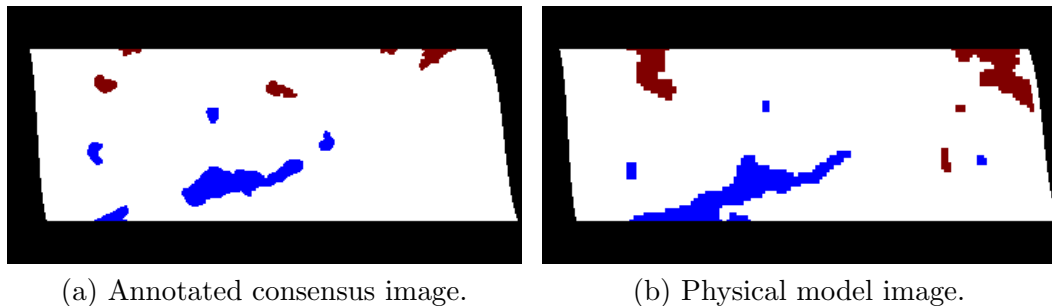
Figure 3.2: Visual clustering interface between the consensus map and a physical model image. Positive polarity coronal holes are red. Negative coronal holes are blue. The date is 07/23/2010.

3.2 Manual coronal hole matching and physical map classification

This section discusses manual classification protocols designed to address the issues discussed in Sec 3.1 while maintaining reproducibility. Before displaying to user, model and consensus maps are preprocessed to have the same size. Each coronal hole is color coded based on its polarity (see Fig 3.2). Furthermore, we removed regions of no observations, and remove regions with latitude that was below 30° and above 150°. The rest of the section describes how to cluster coronal holes, support matching between maps, and the final, manual classification.

3.2.1 Manual coronal hole matching using clustering

In this section, we provide a summary of the coronal hole matching algorithm. We begin with the clustering rules. After clustering, we label each cluster as matchable, new (generated), or missing (removed).

For coronal holes of the same polarity, we demonstrate the clustering rules in Fig.

3.3 and summarize below:

CR1. Cluster polar coronal holes: Coronal holes that were cut at a latitude of 30° are clustered together into the north polar coronal hole cluster. Similarly, coronal holes that were cut at a latitude of 150° are clustered together into the south polar coronal hole cluster.

CR1. Nearby clustering: Coronal holes that extremely close to each other are clustered together.

CR3. Small-small clustering: Groups of small coronal holes that are relatively close to each other are clustered together.

CR4. Large-small clustering: A small coronal hole that is close to a much larger one is considered part of the larger cluster that involves the larger coronal hole.

CR5. No large-large clustering: In general, larger coronal holes are not clustered together unless they are extremely close to each other (see **CR2**).

Coronal hole clusters of the same polarity are matched based on the following rules (see Fig. 3.4):

M1. Polar to polar matching: Polar clusters with a relatively large area overlap (70% to 100%) are matched.

M2. Polar to mid-latitude matching: A coronal hole cluster from the consensus map that is located in the mid-latitude region is matched to a polar cluster from the physical model when they overlap by at-least 15% to 20%, or more.

M3. Mid-latitude to mid-latitude matching: Mid-latitude clusters are matched with good area overlap (e.g., overlap area > 30%) or weaker area overlap but good localization.

After applying the rules, the remaining coronal hole clusters are classified as either new (generated in the model) or removed (missing from the model) (see Fig. 3.4).

(a) **CR1. Cluster polar coronal holes.** (b) **CR1. Cluster polar coronal holes.**

(c) **CR2. Nearby clustering.** (d) **CR3. Small-small clustering.**

(e) **CR4. Large-small clustering.** (f) **CR5. No large-large clustering.**
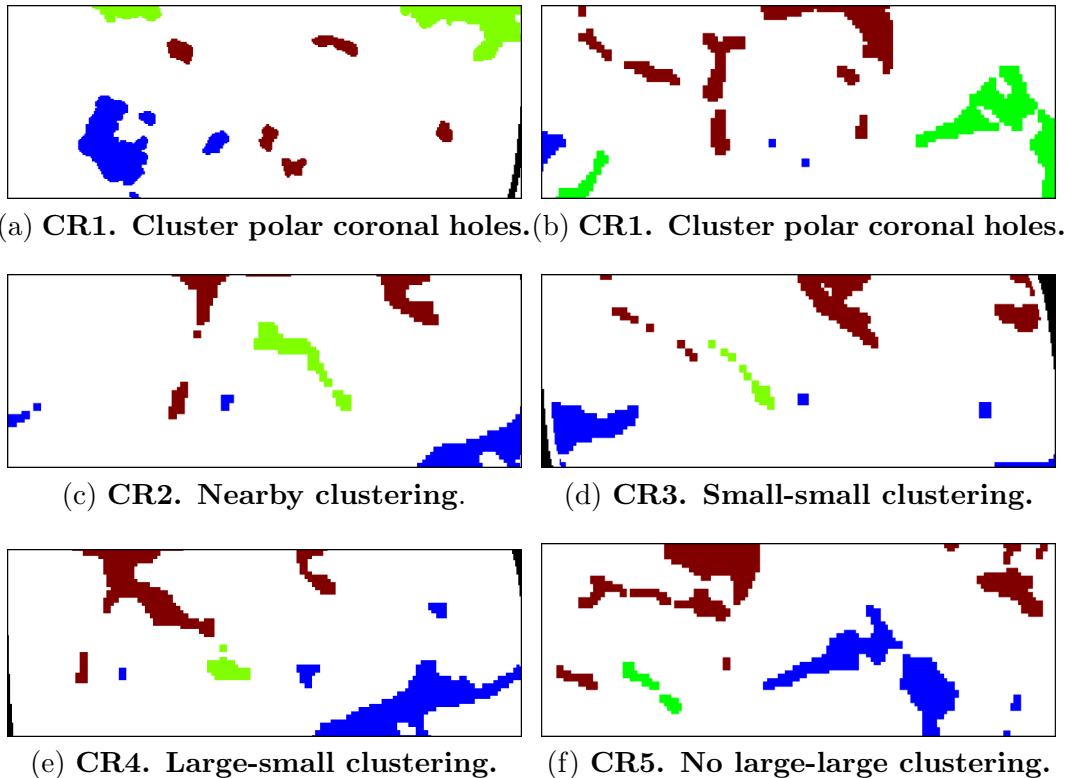
Figure 3.3: Coronal holes clustering rules. The rules are applied to the coronal holes depicted in green.

3.2.2   Map Classification

To support consistency and reproducibility, maps are pre-classified into two groups. We use ranks to describe each group. In the rank 1 group, we include maps that tend to be closer to the consensus map. In the rank 2 group, we include maps that tend to be further away from the consensus map. We then make the final classifications of what constitutes a good and a bad map based on Fig. 3.5.

To decide the rankings, we examine the mid-latitude coronal holes. Initially, similar to clustering, we group maps based on how similar they are to each other. The collection of all of the groups are then classified as being closer to the consensus map (rank 1) or further from the consensus map (rank 2). Here, we classify a group as being closer to the consensus map if it contains a substantial number of matched, fewer cases of new (generated) and missing (removed) coronal holes. A ranked group of maps (rank 1 or 2) is then classified a *good match* if it is in good agreement of the consensus map, where we also allow slight over-estimation of the area of the coronal holes. A group of maps that is not considered a *good match* is classified as a *bad match*.

We present a classification example in Fig. 3.6. We begin by explaining the interface. The filled regions represent coronal hole clusters in the consensus map. The hollow regions represent coronal hole clusters in the physical map. In the interface, a rectangular region (not shown here) was used to specify a matching. Once the matching has been specified, all coronal hole clusters that overlapped with the user-specified rectangular region are selected as a match. A green bounding box is drawn around the matched coronal hole clusters. Note that since the bounding boxes are extended out to the full size of each coronal hole cluster, they also contain unmatched coronal holes. In fact, matched coronal holes are represented with faded colors (not bright blue or red). The remaining coronal hole clusters are automatically classified as new (generated) or removed (missing). They are represented using bright red (positive) or bright blue (negative).

We show examples from two groups in Fig. 3.6. Group 1 maps do not have a matching for the positive polarity coronal hole located in the upper-right region (depicted as bright red). Group 2 maps do have a mathing cluster for the same coronal hole (depicted as faded red). Furthermore, group 2 maps missed (removed) fewer coronal holes (depicted as solid blue here). Thus, group 2 maps are thus

classified as rank 1 and group 1 maps are classified as rank 2. Furthermore, rank 1 maps were classified as *good matchings* and rank 2 maps were classified as *bad matchings*.
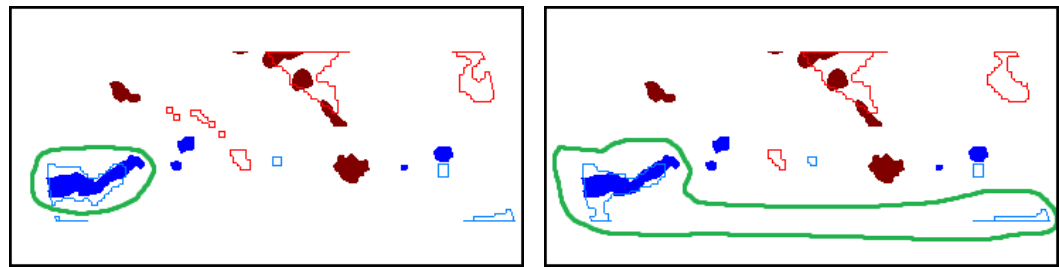
(a) **M1. Polar to polar matching.**

(b) **M1. Polar to polar matching.**
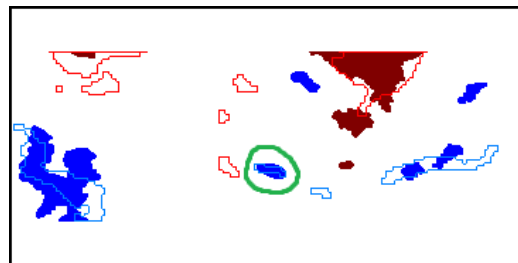
(c) **M3. Mid-latitude to mid-latitude matching with good area overlap.**
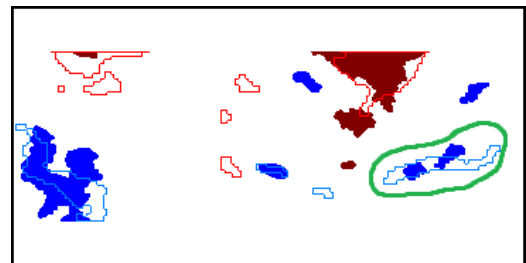
(d) **M3. Mid-latitude to mid-latitude matching with weak area overlap but good localization.**
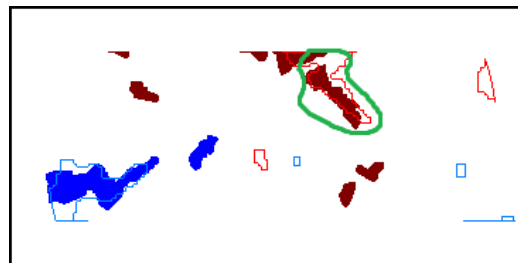
(e) **M2. Polar to mid-latitude matching.**
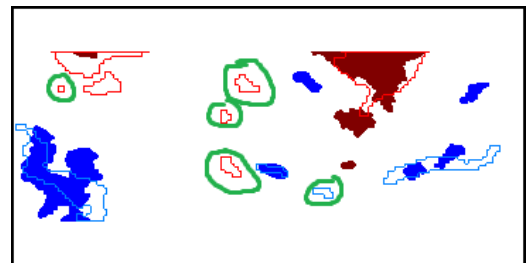
(f) Generated (new) coronal holes
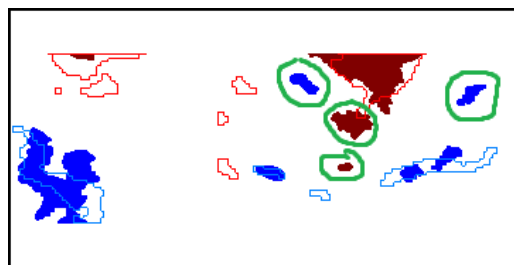
(g) Removed (missing) coronal holes

Figure 3.4: Coronal hole cluster matching. The regions in green demonstrate the rules. Hollow blue regions represent negative coronal hole clusters in the model. Hollow red regions represent positive coronal hole clusters in the model. Solid blue regions represent negative coronal hole clusters in the consensus map. Solid blue regions represent positive coronal hole clusters in the consensus map.

1: ***Group*** maps based on matched clusters
2: ***Rank*** groups as 1 (better) or 2 (worse).

3: **if** (both ranks represent good matches) **then**
4:      ***Classify*** all maps of the day as good matches.
5: **end if**

6: **if** (both ranks represent bad matches) **then**
7:      ***Classify*** all maps of the day as bad matches.
8: **end if**

9: **if** (rank 1 is acceptable and not rank 2) **then**
10:      ***Classify*** maps of rank 1 as good matches.
11:      ***Classify*** maps of rank 2 as bad matches.
12: **end if**

Figure 3.5: Physical map classification based on group ranking.

(a) Model 1, group 1, rank 2.

(b) Model 6, group 1, rank 2.

(c) Model 8, group 1, rank 2.

(d) Model 9, group 1, rank 2.

(e) Model 11, group 1, rank 2.

(f) Model 4, group 2, rank 1.

(g) Model 10, group 2, rank 1.

(h) Model 12, group 2, rank 1.
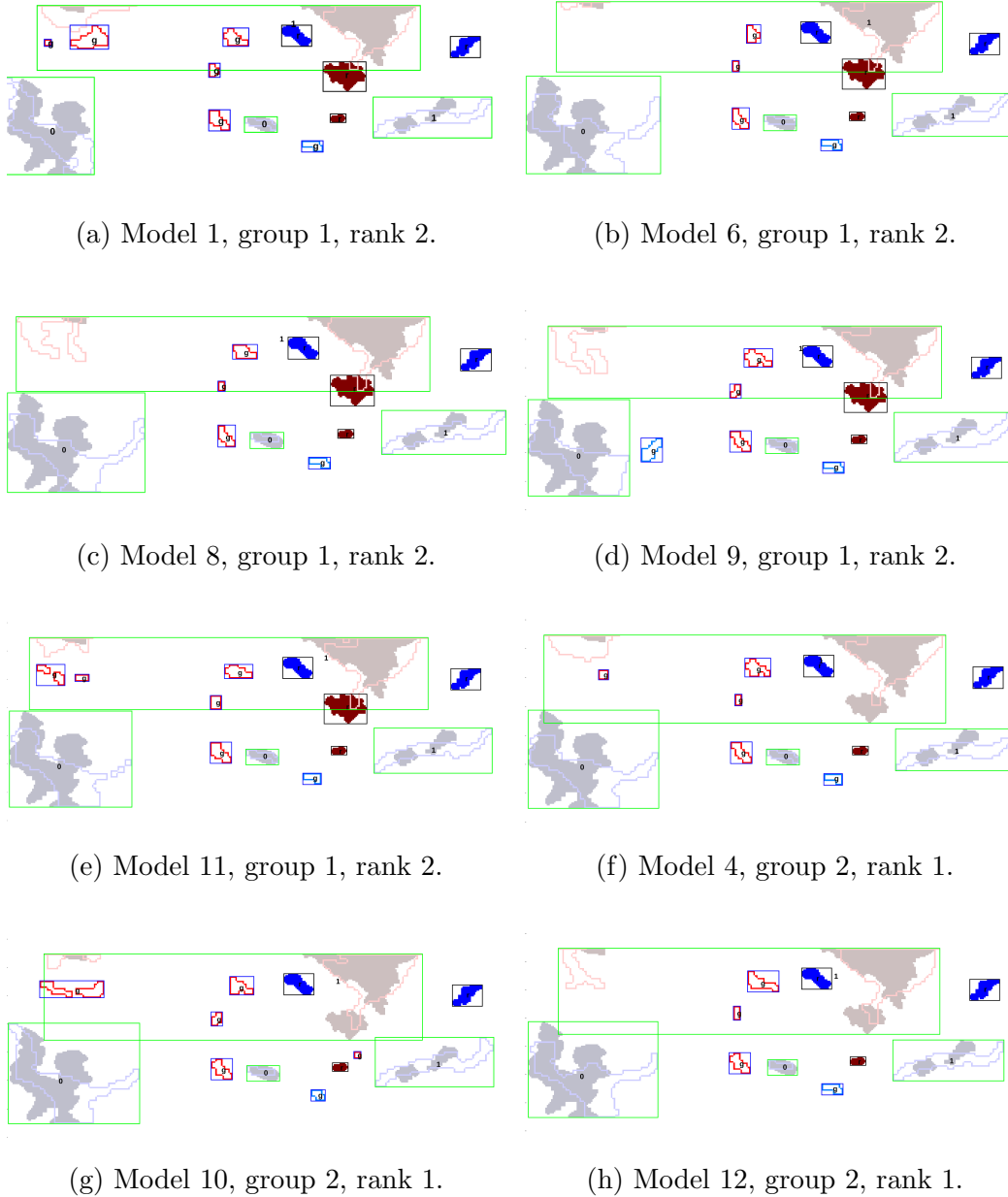
Figure 3.6: Map classification example. We show 8 of the 12 physical maps. Initially, the maps are grouped into two groups (1 and 2). Each group is then assigned a rank (1 or 2). The results refer to the maps associated with 1/20/2011.

Chapter 4

Computer segmentation and classification of physical models

4.1   Overview

4.2   Level set method

We develop a new segmentation method that is based on the Distance Regularized Level Set Evolution (DRLSE) method described in [28]. DRLSE defines:

$$d_p(s) \triangleq p'(s)/s \quad \text{is a divergence operator,}$$

$$g = 1/(1 + |\nabla G * I|^2) \quad \text{is the edge function, and}$$

$$\delta_\epsilon(x) \quad \text{is zero for } |x| > \epsilon \text{ and non-zero for } |x| < \epsilon$$

where $p(.)$ is used for defining a regularized distance for the level set function (ϕ), $g(.)$ should be minimized at image edges, ∇G denotes the gradient of the input image that is computed using convolution with a derivative of a Gaussian. The segmented

image is computed by evolving the level set as given by:

$$\frac{\partial \phi}{\partial t} = \mu \mathcal{R}_p(\phi) + \lambda \mathcal{L}_g(\phi) + \alpha \mathcal{A}_g(\phi) \tag{4.1}$$

where:

$$\mathcal{R}_p(\phi) = \text{div}(\text{d}_\text{p}(|\nabla \phi|)\nabla \phi) \quad \text{is the distance term,}$$

$$\mathcal{L}_g(\phi) = \delta_\epsilon(\phi)\text{div}(\text{g}\frac{\nabla \phi}{|\nabla \phi|}), \quad \text{is the boundary term, and}$$

$$\mathcal{A}_g(\phi) = g\delta_\epsilon(\phi) \quad \text{is an area term.}$$

We provide a description of the proposed segmentation algorithm in Figs. 4.1 and 4.2. The approach requires joint processing of the EUVI and magnetic images. Most importantly, we need to modify the edge function so that it does not allow crossing the magnetic neutral lines. This is accomplished by modifying the edge function to be:

$$\text{pg} = (1 - \text{p})\text{g} \tag{4.2}$$

where p assumes the value of 1 over the magnetic polarity boundaries detected in the magnetic image and is zero away from the boundary (see Fig. 4.1). Thus, over the magnetic lines, the edge function becomes zero and prevents crossing of the neutral line boundary.

From (4.1), we have found that α and the spatial spread of the Gaussian (σ) used for computing the edge function are the two parameters that can affect overall segmentation performance. To find the optimal parameter values, we compare against the consensus maps, and look for the optimal values using (e.g., see [3]):

$$\min_{\alpha, \sigma} \sqrt{[1 - \text{spec}(\alpha, \sigma)]^2 + [1 - \text{sens}(\alpha, \sigma)]^2} \tag{4.3}$$

where Spec denotes the (pixel-level) specificity and Sens denotes the corresponding sensitivity. The solution of (4.3) gives the optimal values for each image. For each

function SEGMENT(EUVI, mag, α, σ)
$I \leftarrow$ **smooth** `syn_img` with 15×15 Gaussian kernel
     with optimization variable σ.
g $\leftarrow \frac{1}{1+(I_x^2+I_y^2)}$

▷ Make g zero at magnetic boundaries
p $\leftarrow$ **DetectMagneticCrossLines**(`mag_img`)
pg $\leftarrow$ (1 - p) .* g

▷ Initialize with HenneyHarvey segmentation method
$\phi_0 \leftarrow$ **HenneyHarvey** (`syn_img`, `photo_img`)

▷ Run with modified edge function
▷   and optimization parameter α.
return LS(EUVI, mag, ϕ_0, pg, α)
end function

Figure 4.1: Main coronal hole segmentation algorithm. The parameters α and σ are optimized using `Pattern-Search`.

image, we constrain the optimization problem [2, 35, 24, 29, 19] for $\alpha \in [-3, +3]$, $\sigma \in [0.2, 1]$. Over the training set, we select the median values over the entire set. We then report the performance over the testing set. In the results section, we report the performance of the algorithm using leave-one-out.

The optimization of (4.3) is challenging since derivative estimates can be very noisy. To this end, we use a robust optimization method based on `Pattern-search` initialized with $\alpha_0 = 0, \sigma_0 = 0.5$. We refer to [30] for details on the optimization procedure. Furthermore, to speed-up convergence, we initialize the segmentation algorithm using the Henney-Harvey algorithm as documented in Fig. 4.1.

function LS(EUVI, mag, ϕ_0, pg, α)
    $\phi \leftarrow \phi_0$,   ▷ init. using previous method
    for i $\leq$ n **do**
        $\delta(\phi) \leftarrow$ **Dirac**(ϕ, ϵ)

        ▷ Use modified edge function pg:
        $F_a \leftarrow$ **areaTerm**$(\delta(\phi),$ pg$)$
        $F_e \leftarrow$ **edgeTerm**$(\delta(\phi), \phi,$ pg$)$
        $F_d \leftarrow$ **Regularize_distance**(ϕ)

        ▷ Allow α to vary for optimization:
        $\phi \leftarrow \phi +$ ts$\cdot(\mu F_d + \lambda F_e + \alpha F_a)$
    end for
  return $\delta(\phi)$
  end function

Figure 4.2: Level-set segmentation algorithm using the modified edge function pg and α.

4.3 Classification system overview

An outline of the algorithm is provided in Fig. 4.3. The algorithm accepts the set of physical maps and a reference map that corresponds to the physical observations. Here, for the best results, the reference map should be the consensus map. Alternatively, we can set the reference map to a manual or an automated segmentation map and measure its performance against the consensus map.

The main algorithm accepts the dates that need to be processed. Then, for each date, we load the physical models and the reference map. All of the maps are pre-processed prior to matching. Very near coronal holes are clustered and a comparison between each physical model and the reference map reveals new and missing coronal hole clusters that are stored in new maps, according to their polarity. The remaining coronal hole clusters are matched using Linear Programming. Features extracted from all maps then used for the final classification.

```
 1: function MAP_CLASSIFICATION(dates)
 2: ▷ Input: dates to process.
 3: ▷ Classify physical maps for given dates.
 4:
 5:       ▷ Process each date separately
 6:    for date ∈ dates do
 7:        ▷ Read and process reference image
 8:        ▷ (e.g., Consensus image, or automatically segmented images).
 9:        ref_map ← load_ref_data(date)
10:        ref_map_{+,−} ← pre_process(ref_map)
11:
12:         ▷ Process associated physical models
13:        for model ∈ {model_1,...,model_12} do
14:            model_map ← load_model(date, model)
15:            model_polarity_map_{+,−} ← pre_process(mod_img)
16:
17:             ▷ Analyze each polarity separately
18:            for polarity p ∈ { +, − } do
19:                Cluster coronal holes that are are very close.
20:                Detect coronal hole clusters that are in
21:                     physical maps but not in reference map
22:                     using Mahalanobis distance threshold and
23:                     store the results in new_map_p and missing_map_p
24:                Re-cluster remaining coronal holes in ref_map and
25:                     model_map to achieve equal number of clusters.
26:                Match clusters using linear programming
27:                     and save the results in matched_map_p
28:            end for
29:            Extract features from new_map_p, missing_map_p
30:                 and matched_map_p for polarity p ∈ {+, -}.
31:            Classify model using extracted features.
32:        end for
33:    end for
34: end function
```

Figure 4.3: Overview of physical map classification algorithm.

The rest of the chapter is organized into four subsections. In section 4.4, we describe the pre-processing steps. Section 4.5 describes clustering of very close coronal

holes. Section 4.6 describes the process for detecting new and missing coronal hole clusters. The matching process which involves re-clustering is described in section 4.7. The classification step is described in section 4.8.

4.4   Map preprocessing

The pre-processing steps are listed in Fig. 4.4. The basic steps involve resizing the maps to the same resolution, removing regions near the poles and regions where we have no observations, and splitting the maps into positive and negative polarity maps.

4.5   Clustering very close coronal holes.

Model maps could contain coronal holes that are very close. Processing these coronal holes individually may lead to improper working of **Detection** and **Matching** algorithms. Hence we cluster very near coronal holes. Clustering is based on minimum pixel distance between coronal holes. The basic idea is to cluster coronal holes iteratively till there are no more coronal holes which are seperated by a threshold.

4.6   Detecting new and missed coronal hole clusters

A coronal hole cluster that is present in the reference map may be *missing* from the physical model map. Alternatively, a physical model may have *new* coronal hole clusters that are absent from the reference map. Clearly, *new* and *missing* coronal

```
 1: function PRE_PROCESS(maps)
 2:      ▷ Pre-process maps to remove regions of no observations, polar regions, and
 3:      ▷  split them based on polarity.
 4:
 5:      ▷ Extract reference and model maps
 6:      [ref_map, model_map] ← extract_coronal_hole_maps (maps)
 7:
 8:      ▷ Remove small gaps in the maps
 9:      [ref_map, model_map] ← binary_close (ref_map, model_map)
10:
11:      ▷ Resize maps to photomap image size
12:      [ref_map, model_map] ← resize_to_same_size (ref_map, model_map)
13:
14:      ▷ Extract regions based on polarity in magnetic images
15:      [ref_magnetic, model_magnetic] ← extract_magnetic_maps (maps)
16:      ref_map₊,₋ ← extract_polarity_maps (ref_map, ref_magnetic)
17:      model_map₊,₋ ← split_coronal_maps (model_map, model_magnetic)
18:
19:      ▷ Remove regions where there are no observations,
20:      ▷  latitude 0 to 30 degrees, and 150 to 180 degrees
21:      no_data_map ← set_no_obs_regions (maps)
22:
23:      ▷ Remove no data regions from all maps
24:      ref_map₊,₋ ← remove_no_data (ref_map₊₋, no_data_map)
25:      model_map₊,₋ ← remove_no_data (mod_map₊,₁, no_data_map)
26: end function
```

Figure 4.4: Pre-processing reference maps and model maps prior to analysis.

hole clusters are to be removed from Both maps, prior to matching. In this section, we describe how to detect them.

To detect *new* coronal hole clusters, we examine each coronal hole cluster in the physical model and find the nearest corresponding one in the reference map. Then, we compute the Mahalanobis distance between the origin (perfect match) and the vector composed of the minimum physical distance between the coronal holes and their physical area difference. If the Mahalanobis distance, the coronal hole in the

physical model is classified as new.

Similarly, to detect *missing* coronal hole clusters, we examine each coronal hole cluster in the reference map and look for the corresponding one in the physical map. Then, a coronal hole cluster in the reference map is classified as *missing* if is has a high Mahalanobis distance from the origin.

4.7   Matching with re-clustering

After removing the new and missing coronal hole clusters, the remaining ones need to be matched. Unfortunately, we can still have different numbers of clusters in each map. Thus, instead of matching maps having different number of clusters, we need to first combine them together to have equal numbers of clusters. Clustering is accomplished using the minimum physical distance between coronal holes. The basic idea is to iteratively cluster together all coronal hole clusters that are separated by a minimal physical distance until we reach the desired number of clusters. After clustering we introduce linear programming model for computing an optimal matching between coronal hole clusters.

Let i be used to index clusters in the reference map. Similarly, let j be used to index clusters in the physical map. Then, we use $m_{i,j}$ to denote a possible match between cluster i in the reference map and cluster j in the physical map. Thus, $m_{i,j} = 1$ when there is a match between the clusters and $m_{i,j} = 0$ otherwise. We also assign a cost $w_{i,j}$ associated with the matching. Here, we set the $w_{i,j}$ to be the shortest spherical distance between the clusters. Thus, $w_{i,j} = 0$ when the clusters overlap.

Formally, we find an optimal matching by solving:

$$\min_{m_{i,j}} \sum_i \sum_j w_{i,j} m_{i,j} \tag{4.4}$$

subject to:

$$\sum_i m_{i,j} = 1, \tag{4.5}$$

$$\sum_j m_{i,j} = 1 \tag{4.6}$$

$$m_{i,j} \in \{0, 1\} \tag{4.7}$$

where $m_{i,j}$ denotes the assignment that minimizes the weighted matching of (4.4), while each cluster can only be assigned to one other cluster as required by (4.5) and (4.6). This is a typical bipartite matching setup, making matching matrix created from $m_{i,j}$ to be totally unimodular. As discussed in [31] this problem when solved with linear programming will return an integer solution.

4.8   Classification

Each physical map is finally classified as *good* or *bad* (see section 3.2.2) The following features are used for classification:

- **Number of new coronal holes**: Number of coronal holes that are predicted by model but absent in reference map.

- **Number of missing coronal holes:** Number of coronal holes that are missing from model but present in reference map.

- **Total area of new coronal holes**: Total area of generated coronal holes projected onto an unit sphere.

- **Total area of missing coronal holes:** Total area of removed coronal holes projected onto an unit sphere

- **Over estimated area**: Area overestimated by model.

Following feature extraction, we apply principal component analysis to reduce the features. Classification is performed using k-nearest neighbors (KNN) $K = 11$ and SVM.

Chapter 5

Results

5.1   Summary

The results are summarized in six sections. Section 5.2 provides a description of the dataset. Section 5.3 provides results for coronal hole segmentation. Section 5.4 provides clustering examples. Section 5.5 summarizes results for coronal hole detection. Pre-clustering and matching between physical and consensus maps are given in 5.6. Final classification results are given in section 5.7.

5.2   Classification dataset

The dataset consisted of two Carrington rotations [23] that consisted of 50 days. The first Carrington rotation covers the dates from 07/13/2010 to 08/09/2010. The second Carrington rotation covers the dates from 01/20/2011 to 02/16/2011. For each day, we have

- Synoptic image [4]

- Magnetic photo map image

- Two human segmentations

- Consensus map derived from the two human segmentations

- Twelve coronal hole prediction maps

Consensus maps are used to manually determine good and bad prediction maps. For comparing to human performance, we also compare the results against the use of two human segmentations (human raters labeled R4 and R7). For training, we use leave-one-out over 10 randomly chosen dates and report results on the remaining 40.

5.3   Segmentation

We present results for the consensus maps ($N = 46$) in Table 5.1. On average, we have a reduction of 19% ($\sigma = 17.7\%$) in the unit-distance from the ideal segmentation. On the other hand, we also have cases where the new method performs worse than the original `Henney-Harvey` method. We provide three representative examples in Figs. 5.1, 5.2, and 5.3.

The best case scenario is shown in Fig. 5.1. Here, it is clear that the level-set method provides smoother boundaries with coronal hole estimates that are better filled and in better agreement with the `Henney-Harvey` method. On the other hand, there are significant gaps in the original method.

Similar comments apply for the typical case shown in Fig. 5.2. Overall, a larger number of coronal holes appear in the original method that cannot be found in the consensus map. On the other hand, most of the smaller coronal holes are missing from the level-set approach.

For both methods, the worst case is shown in Fig. 5.3. In this example, both methods fail to fully detect the coronal hole in the north pole region. A careful examination of the EUVI image of Fig. 5.3(a) shows that the pixels in this north-pole coronal hole are much brighter than average. Thus, the initialization by the `Henney-Harvey` method fails to detect the coronal hole in the south pole and the level-set evolution does not recover from this initialization. Furthermore, the consensus map appears to have finer resolution detail than both maps. In this case, the level-set approach overly smooths the detected components as compared to the consensus map and the original method.

Table 5.1: Percentage improvement of proposed level-set segmentation method (see (4.3) for definition of unit distance). Overall, we have an average (mean) improvement of **19.01**% with a standard deviation of **17.7**%

Order Stat.	Henney-Harvey	Level Sets	% Impr.
Min	0.14	0.17	-24.96
25%	0.26	0.27	-2.00
50%	0.31	0.27	12.06
75%	0.23	0.16	30.44
Max	0.19	0.09	50.76

5.4   Clustering

Coronal holes which are very close to each other are clustered together based on pixel distance. In this section clustering in reference map and model map are demonstrated in Fig. 5.4 and Fig. 5.5. Clustered coronal holes are filled with same shade of red or blue. Shades of red is used to mark positive clusters while blue marks negative clusters. As seen from Fig. 5.5 consensus maps rarely have very near coronal holes.

5.5 Detection of new and missing coronal holes

In order to detect new and missing coronal holes, the distance threshold parameter was experimentally set on the ten randomly chosen dates. Then, independent testing on the remaining 40 dates was 87.7%.

We present two detection examples. An example where everything worked well is shown in Fig. 5.6. On the other hand, a difficult case is shown in Fig. 5.7. Visual inspection of the coronal holes of Fig. 5.6 demonstrates that the physical model map matched the consensus map and did not produce a large number of new and missing coronal holes. On the other hand, there was much more activity and there were many differences in the maps of Fig. 5.7.

5.6 Matching

Results of matching using linear programming are demonstrated in Figs. 5.8 and 5.9. Fig. 5.8 shows an example where automated matching agrees with manual matching. Fig. 5.9 shows an example where there are significant differences between automated and manual matching (see pink cluster).

5.7 Classification results

Map classification is performed using KNN and SVM classifiers. Classifiers are trained on 10 randomly chosen dates and tested on remaining 40 (as described in Sec 5.2). Table 5.2 provides a side by side comparison of classification accuracy when using consensus and automatically segmented maps as references. We can see that the use of automatically segmented maps produced results that approximated the

|	Consensus Vs model				Auto segmentation Vs model			
	KNN		SVM		KNN		SVM	
	good	*bad*	*good*	*bad*	*good*	*bad*	*good*	*bad*
good	143	46	29	36	121	12	111	15
bad	84	207	198	217	106	241	116	238
Accuracy	72.9%		51.2		75.4%		72.7%	

Table 5.2: Classification results 1

	R4 Vs model				*R7* Vs model			
	KNN		SVM		KNN		SVM	
	good	*bad*	*good*	*bad*	*good*	*bad*	*good*	*bad*
good	78	88	55	33	74	39	34	15
bad	149	165	172	220	141	214	181	238
Accuracy	50.6%		57.2%		61.5%		58.1%	

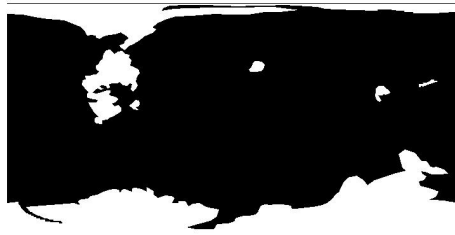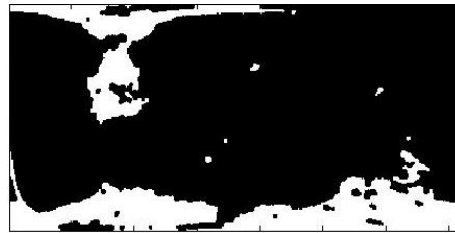Table 5.3: Classification results 2

use of consensus maps.

In table 5.3, we present results based on the use of the original, manual segmentations (*R4* and *R7*). We can see that *R7* is closer to consensus maps than *R4*.
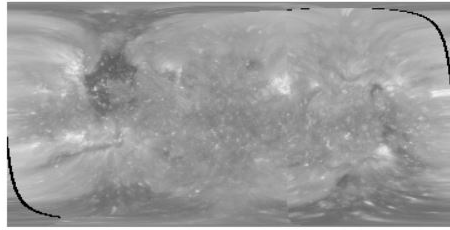
(a) EUVI
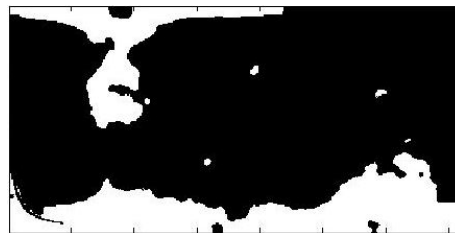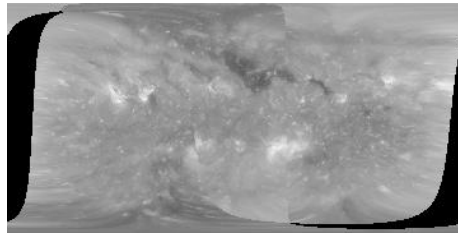

(b) Consensus map

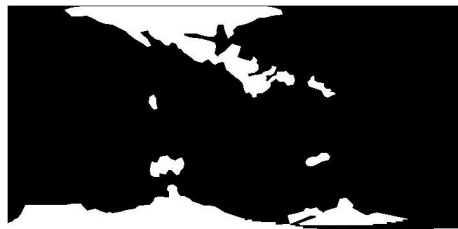(c) `Henney-Harvey` Method, `unit_dist` $= 0.19$

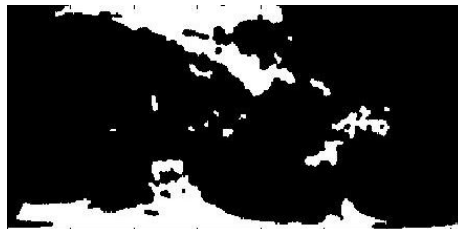(d) Level-set segmentation, `unit_dist` $= 0.09$

Figure 5.1: Best case results for the level-set segmentation method for the input data from January, 24th, 2011. We have a reduction in the unit distance by 50.76% (see (4.3) for definition of unit distance).

(a) EUVI

(b) Consensus map

(c)   `Henney-Harvey`   Method,
`unit_dist` $= 0.17$

(d)   Level   set   segmentation,
`unit_dist` $= 0.12$

Figure 5.2: Typical case results for the level-set segmentation method for the input data from May 2, 2011. We have a reduction in the unit distance by 29.36% (see (4.3) for definition of unit distance).

(a) EUVI

(b) Consensus map

(c)   `Henney-Harvey`   method,
`unit_dist` = 0.14
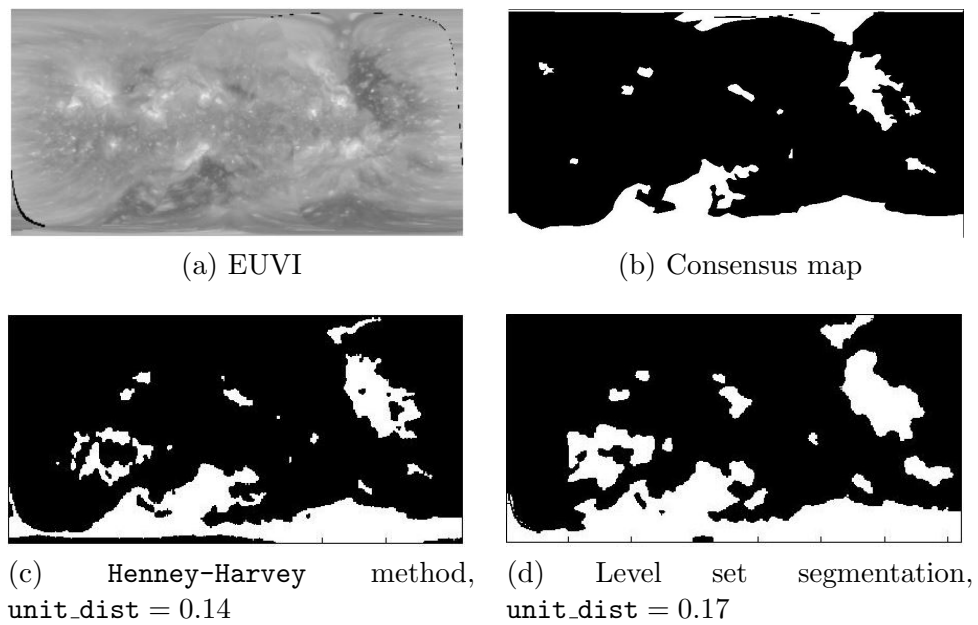
(d)   Level   set   segmentation,
`unit_dist` = 0.17

Figure 5.3: Worst case results for the level-set segmentation method for the input data from February 29, 2011. In this case, we have an increase in the unit distance by 24.96% (see (4.3) for definition of unit distance).

(a) Positive model map before clustering

(b) Positive model map after clustering closest coronal holes

(c) Negative model map before clustering

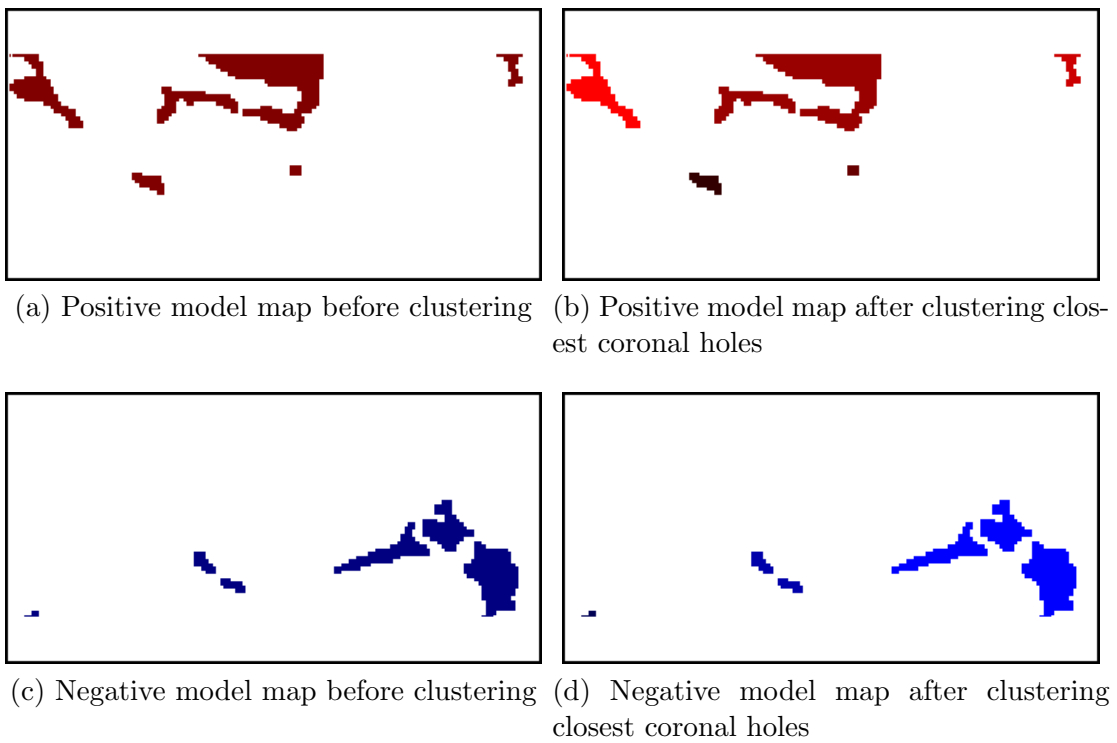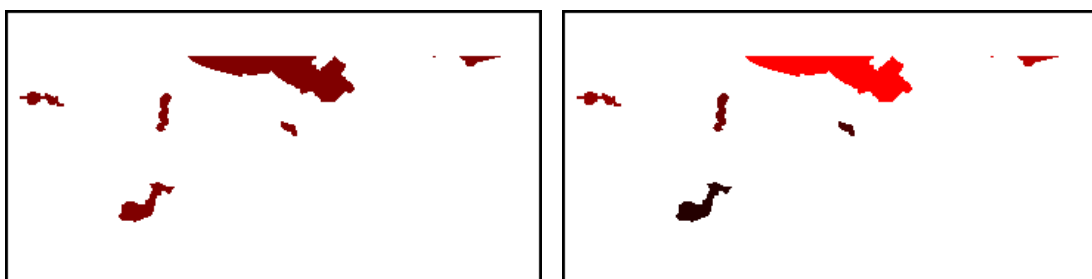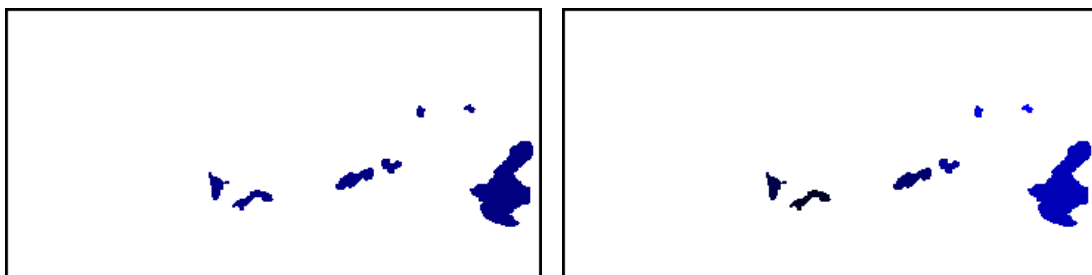(d) Negative model map after clustering closest coronal holes

Figure 5.4: Clustering model

(a) Positive consensus map before clustering

(b) Positive consensus map after clustering closest coronal holes

(c) Negative consensus map before clustering

(d) Negative consensus map after clustering closest coronal holes

Figure 5.5: Clustering consensus

(a) Consensus map

(b) Model map

(c) Coronal holes missing from model map.

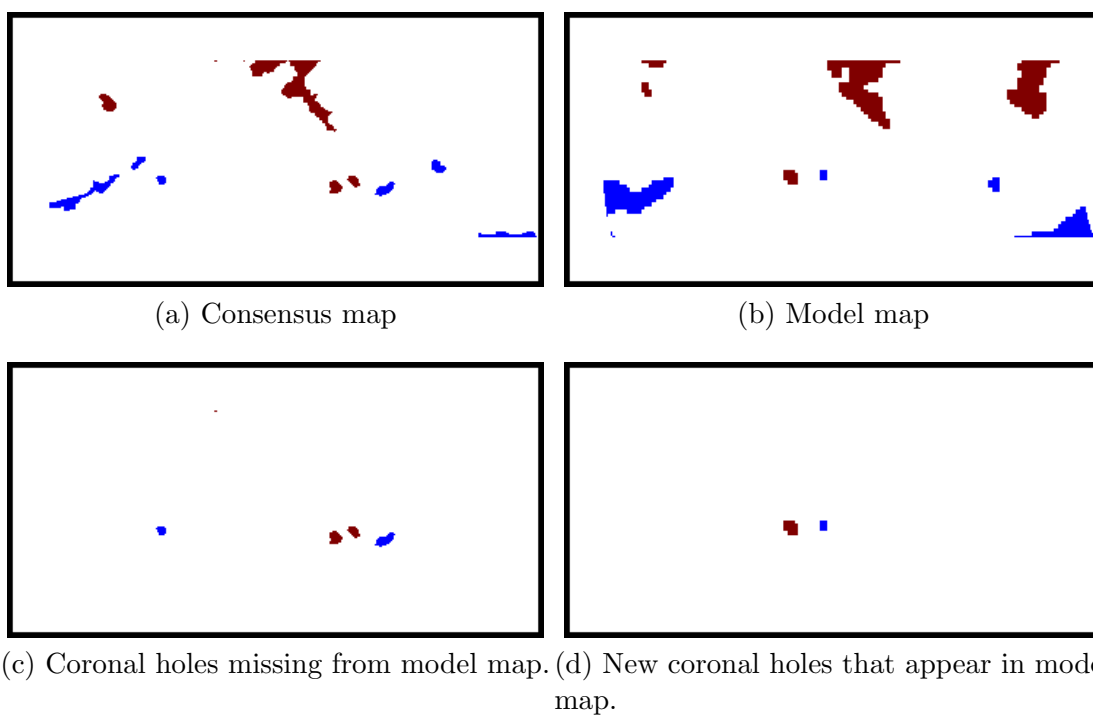(d) New coronal holes that appear in model map.

Figure 5.6: An example that demonstrates good detection of missing and new coronal holes (07-13-2010). In this case, manual labeling and the algorithm agreed on all of the coronal holes except for the upper-right coronal hole depicted in (d).

(a) Consensus map.

(b) Model map.

(c) Coronal holes missing from model.

(d) New coronal holes in model.
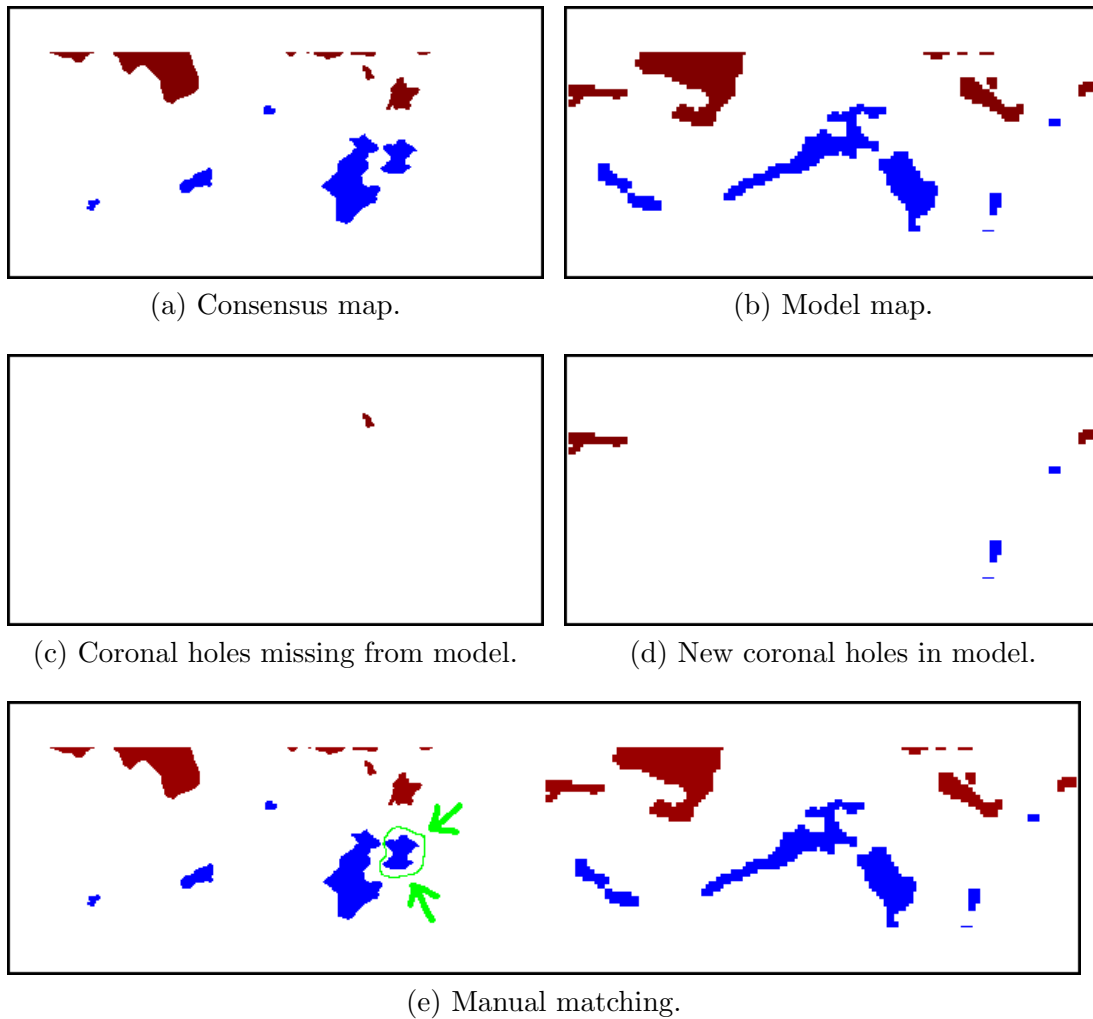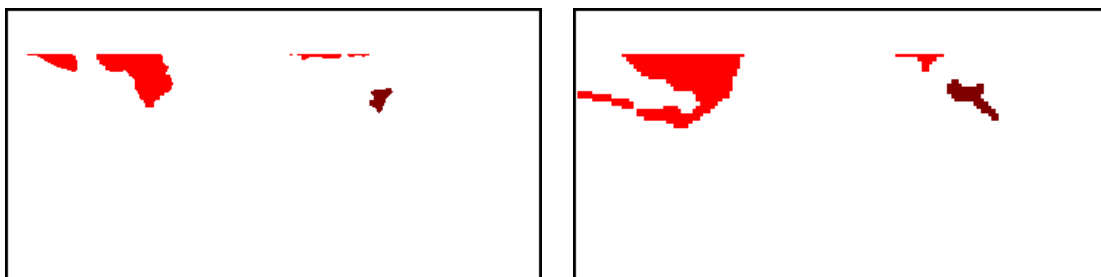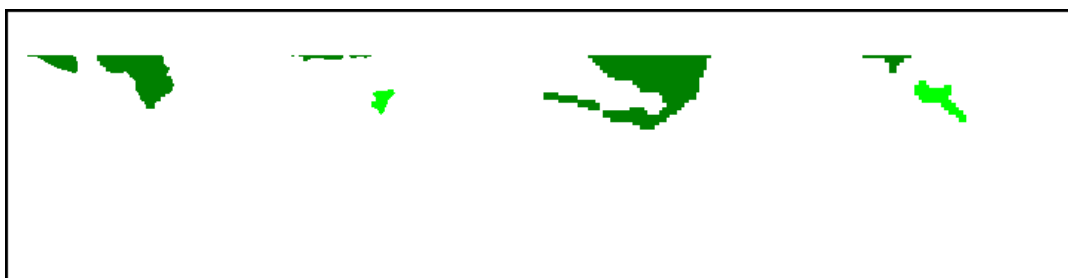

(e) Manual matching.

Figure 5.7: A difficult coronal hole detection example (05-02-2011). The coronal hole shown in green is manually classified as removed but it is classified as matched by the algorithm.

(a) Clustered, positive polarity consensus map.

(b) Clustered, positive polarity model map.

(c) Matching clusters from Consensus (left) and model (right) for positive polarity.

(d) Clustered, negative polarity consensus map.

(e) Clustered, negative polarity model map.

(f) Matching clusters from consensus (left) and model (right) for negative polarity.

Figure 5.8: Good matching example (02-04-2011). Matched clusters are shown using the same colors.
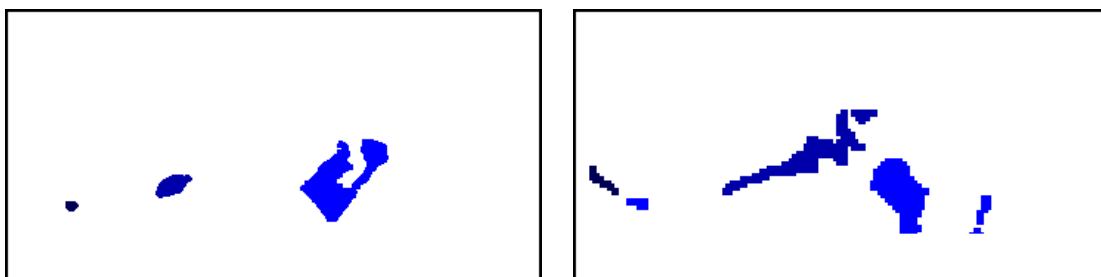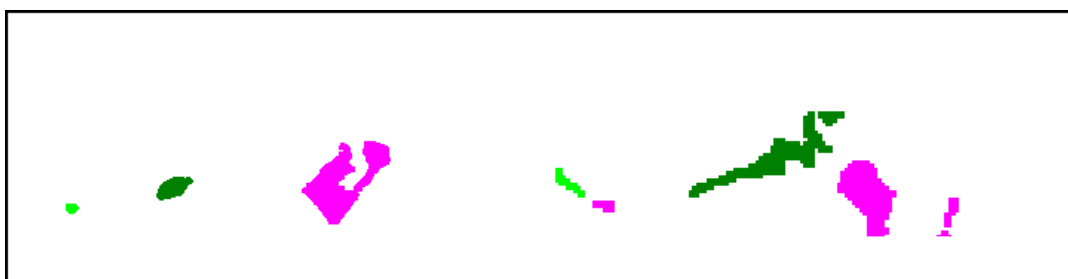
(a) Clustered, positive polarity consensus map.

(b) Clustered, positive polarity model map.

(c) Matching clusters from consensus (left) and model (right) for positive polarity.
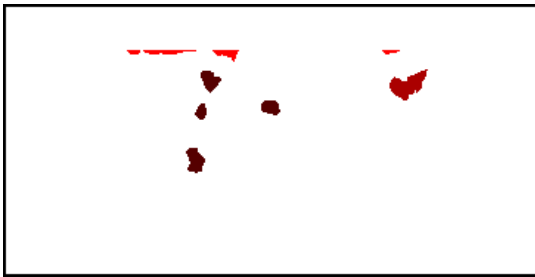

(d) Clustered, negative polarity consensus map.

(e) Clustered, negative polarity model map.

(f) Matching clusters from consensus (left) and model (right) for negative polarity.
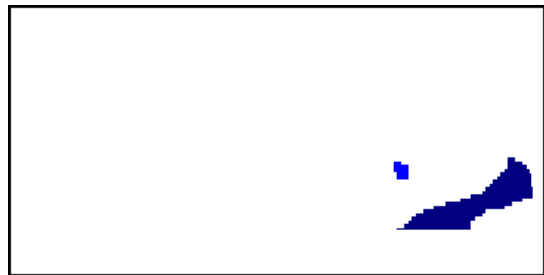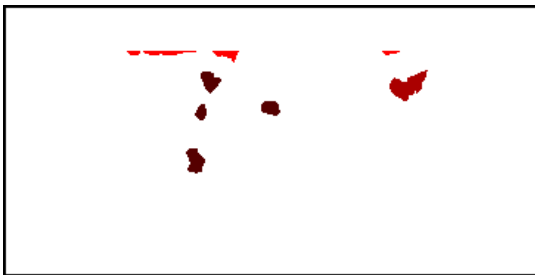
Figure 5.9: Difficult example of coronal hole matching showing issues in the algorithm (07-08-2010). Matched clusters are shown using same colors.

Chapter 6

Conclusion

6.1 Thesis overview

The primary contributions of the thesis include: (i) a new segmentation method to detect coronal holes, (ii) a manual protocol to support reproducible classification of physical models, and (iii) an automated method for physical model classification. In each case, the performance of each method was validated against human experts. By comparing against the consensus maps, the new coronal hole segmentation method was shown to work better than the currently used method. Similarly, when compared against consensus maps, automated classification of the physical maps has been shown to be equal or better than what can be achieved by individual raters. Here, it is important to note that performance cannot exceed what can be done by the consensus maps, since these maps are used as the ground truth for the segmentation methods.

6.2   Future Work

There is big room for improvement in the proposed methods. A summary of the most important issues includes:

- **Larger number of manual segmentation by independent raters.** There were significant differences between the consensus maps and the individual segmentation maps produced by each rater. Such issues can only be addressed by including a substantially larger number of raters. In this case, if nothing else, the collection of all of the maps that will be produced will likely not missed any critical coronal holes.

- **Inter-rater and intra-rater variability studies.** Based on a larger number of raters, we can measure the statistical variation between them, as well as the variation when the same rater repeats the process.

- **Studies on larger databases.** Clearly, it will be interesting to extend the study to cover more dates.

- **Prediction studies.** Instead of looking at standard solar cycles, it will be interesting to develop prediction methods that use previously processed maps to classify physical maps in future dates.

- **Extracting coronal holes from the Helios events knowledgebase (HEK).** It will be interesting to repeat the study by using HEK maps to generate consensus maps and then repeat the study based on the new consensus maps.

References

[1] *Open Source Computer Vision.* Opencv.org, 2013.

[2] C. Agurto, V. Murray, H. Yu, J. Wigdahl, M. Pattichis, S. Nemeth, S. Barriga, and P. Soliz. A multiscale optimization approach to detect exudates in the macula. *IEEE Journal of Biomedical and Health Informatics*, 18(4):1328–1336, July 2014.

[3] C. Agurto, V.M. Murray, H. Yu, J. Wigdahl, M.S. Pattichis, S. Nemeth, S. Barriga, and P. Soliz. A multiscale optimization approach to detect exudates in the macula. *IEEE Journal of Biomedical and Health Informatics*, 2014.

[4] Martin D Altschuler, Randolph H Levine, Michael Stix, and John Harvey. High resolution mapping of the magnetic field of the solar corona. *Solar Physics*, 51(2):345–375, 1977.

[5] Martin D Altschuler, Dorothy E Trotter, and Frank Q Orrall. Coronal holes. *Solar Physics*, 26(2):354–365, 1972.

[6] SK Antiochos, CR DeVore, JT Karpen, and Z Mikić. Structure and dynamics of the suns open magnetic field. *The Astrophysical Journal*, 671(1):936, 2007.

[7] C.N. Arge. Private communications.

[8] C.N. Arge, K.L. Harvey, H.S. Hudson, and S.W. Kahler. Narrow coronal holes in yohkoh soft x-ray iimage and the slow solar wid. *Solar Wind Ten*, 679, 2003.

[9] C.N. Arge, K.L. Harvey, H.S. Hudson, and S.W. Kahler. Narrow coronal oles in yohkoh soft x-ray iimage and he slow solar wind. *Solar Wind Ten*, 679, 2003.

[10] C.N. Arge, C.J. Henney, J. Koller, W.A. Toussaint, J.W. Harvey, and S.L. Young. Improving data drivers for coronal and solar wind models. In *AS-TRONUM 2010*, volume 444 of *ASP Conference*, pages 99–104, 2011.

References

[11] C.N. Arge, J.G. Luhmann, D. Odstrcil, C.J. Schrijver, and Y. Li. Stream structure and coronal sources o the solar wind during the may 12th, 1997 cme. *Atoms. Sol. Terr. Phys.*, 66:1295–1309, 2004.

[12] C.N. Arge, D. Odstrcil, V.J. Pizzo, and L.R. Mayer. Improved method for specifying solar wind speed near the sun. In M. Velli, F.Malara, and R. Bruno, editors, *Solar Wind Ten*, volume 679, pages 190–193, 2003.

[13] C.N. Arge and V.J. Pizzo. Improvement in the prediction of solar wind conditions using near-real time solar magnetic field updates. *Geophys. Res.,*, 2000.

[14] M.J. Aschwanden. Image processing techniques and feature recognition in solar physics. *Solar Physics*, 262:235–275, 2010.

[15] Saeid Belkasim, A Ghazal, and Otman A Basir. Phase-based optimal image thresholding. *Digital Signal Processing*, 13(4):636–655, 2003.

[16] A.C. Bovik. *The Essential Guide to Image Processing*. Academic Press, 2009.

[17] V. Delouile, B. Mampaey, C. Verbeeck, and R. de Visscher. SPOCA suite: a software for extraction and tracking of active regions and coronal holes on euv images. Technical report, 2012.

[18] S. DeMarco. Vaidating coronal hole detection and solary wind predictions. Technical report, 2012.

[19] G. Esakki, V. Jatla, and M. Pattichis. Optimal hevc encoding based on gop configurations. In *IEEE Southwest Symposium on Image Analysis and Interpretation*, Santa Fe, New Mexico, 2016.

[20] Joseph L. Fleiss. *Design and Analysis of Cinical Experiments*. Wiley Interscience, 1999.

[21] K.L. Harvey and F. Recely. Polar coronal holes during cycles 22 and 23. *Solar Physics*, 211:31–33, 2002.

[22] C.J. Henney and J.W. Harvey. Automated coronal hole detection using he i 1083 nm spectroheliograms and photospheric magnetograms. In *ASP Conf.*, volume 346, pages 261–268, 2005.

[23] Robert Howard and Barry J Labonte. Surface magnetic fields during the solar activity cycle. *Solar Physics*, 74(1):131–145, 1981.

[24] Y. Jiang and M. Pattichis. Jpeg image compression using quantization table optimization based on perceptual image quality assessment. In *Proc. 45th Asilomar Conference on Signals, Systems and Computers*, pages 225–229, Nov 2011.

References

[25] R. A. Johnson and D. W. Wichern. *Applied Multivariate Statistical Analysis.* Prentice Hall, 1992.

[26] Larisza D Krista and Peter T Gallagher. Automated coronal hole detection using local intensity thresholding techniques. *Solar Physics*, 256(1-2):87–100, 2009.

[27] CO Lee, CN Arge, D Odstrčil, G Millward, V Pizzo, JM Quinn, and CJ Henney. Ensemble modeling of cme propagation. *Solar Physics*, 285(1-2):349–368, 2013.

[28] Chunming Li, Chenyang Xu, Changfeng Gui, and M.D. Fox. Distance regularized level set evolution and its application to image segmentation. *IEEE Transactions on Image Processing*, 19(12):3243–3254, Dec 2010.

[29] S. Murillo, M. Pattichis, P. Soliz, S. Barriga, C. Loizou, and C. Pattichis. Global optimization for motion estimation with applications to ultrasound videos of carotid artery plaques. In *Proc. SPIE Medical Imaging: Computer-Aided Diagnosis*, pages 76290X–76290X. International Society for Optics and Photonics, 2010.

[30] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization.* Springer, 2nd edition, 2006.

[31] Christos H Papadimitriou and Kenneth Steiglitz. *Combinatorial optimization: algorithms and complexity.* Courier Corporation, 1982.

[32] Constantinos S. Pattichis, Marios S. Pattichis, and E. Micheli-Tzanakou. Medical imaging fusion applications: An overview. In *Thirty-Fifth Asilomar Conference on Signals, Systems and Computers*, volume 2, pages 1263 – 1267, 2001.

[33] M.S. Pattichis, C.S. Pattichis, M. Avraam, A.C. Bovik, and K. Kyriakou. Am-fm texture segmentation in electron microscopic muscle imaging. *IEEE Transactions on Medical Imaging*, 19(12):1253–1258, 2000.

[34] M.S. Pattichis, P. Soliz, and T. Cacoullos. New models for region of interest reader classification analysis in chest radiographs. *Pattern Recognition, Special Issue of Digital Image Processing and Pattern Recognition Techniques for Detection of Cancer*, 42(6):1058–1066, 2009.

[35] P. Rodriguez-Valderrama, M. Pattichis, and V. Murray-Herrera. System and methods of regularized optimization for matrix factorization and image and video reconstruction, December 9 2014. US Patent 8,908,992.