Astronomical Pipeline Processing Using Fuzzy Logic

Lior Shamir [a,*] Robert J. Nemiroff [b]

[a] *Image Analysis Group, Laboratory of Genetics, NIA/NIH, 333 Cassell Dr., Baltimore, MD 21224, USA*

[b] *Department of Physics, Michigan Tech, 1400 Townsend Dr., Houghton, MI 49931, USA*

Abstract

Fundamental astronomical questions on the composition of the universe, the abundance of Earth-like planets, and the cause of the brightest explosions in the universe are being attacked by robotic telescopes costing billions of dollars and returning vast pipelines of data. The success of these programs depends on the accuracy of automated real time processing of images never seen by a human, and all predicated on fast and accurate automatic identifications of known astronomical objects and new astronomical transients. In this paper the needs of modern astronomical pipelines are discussed in the light of fuzzy-logic based decision-making. Several specific fuzzy-logic algorithms have been develop for the first time for astronomical purposes, and tested with excellent results on a test pipeline of data from the existing Night Sky Live sky survey.

Key words: Fuzzy Logic, Astronomy, Astronomical Pipelines

1 Introduction

In the past few years, pipelines providing astronomical data have been becoming increasingly important. The wide use of robotic telescopes has provided significant discoveries, and sky survey projects are now considered among the premier projects in the field astronomy. In this paper we will concentrate on the ground based missions, although future space based missions like Kepler

* Corresponding author: Tel: (410) 558-8682 Fax: (410) 558-8331
Email address: shamirl@mail.nih.gov (Lior Shamir).

[4], SNAP [19], and JWST [5] will also create significant pipelines of astronomical data.

Pan-STARRS [11], a 60 million dollar venture, is being built today and is expected to be completed by 2006. Pan-STARRS will be composed of 4 telescopes pointing simultaneously at the same region of the sky. Each telescope will be equipped with a 1.8 meter primary mirror and a CCD focal plane mosaic with one billion pixels. With coverage of 6000 degrees^2 per night, Pan-STARRS is expected to observe the entire available sky several times each month, looking for transients that include supernovas, planetary eclipses, and asteroids that might pose a future threat to Earth. Similarly but on a larger scale, ground-based LSST [22] is planned to use a powerful 8.4 meter robotic telescope that will cover the entire sky every 10 days. LSST will cost $200M, be completed by 2012, and produce 13 terabytes per night. Current sky survey projects include SDSS [10, 25], which uses a 2.5 meter robotic telescope trying to map the entire visible universe. In addition, many smaller scale robotic telescopes are being deployed and their number is growing rapidly.

However, in the modern age of increasing bandwidth, human identifications are many times impracticably slow. Therefore, one can reasonably assume that many discoveries of significant scientific value are yet hidden inside the huge databases provided by the deployment of robotic telescopes.

Useful automatic pipeline processing of astronomical images depends on accurate algorithmic decision making. For previously identified objects, one of the first steps in computer-based analysis of astronomical pictures is an association of each object with a known catalog entry. This necessary step enables such science as automatically detected transients and automated photometry of stars. Since computing the topocentric coordinates of a given known star at a given time is a simple and common task, transforming the celestial topocentric coordinates to image (x, y) coordinates might provide the expected location of any star in the frame. However, in many cases slight shifts in the orientation, inaccuracy of the optics or imperfections in the CCD can make this seemingly simple task formidable.

Another essential step in astronomical pipelines is the removal of cosmic ray hits. Except from their annoying presence in astronomical images, cosmic ray hits might be mistakenly detected as true astronomical sources. Algorithms that analyze astronomical frames must ignore the peaks caused by cosmic ray hits, yet without rejecting the peaks of the true astronomical sources. This problem becomes even more significant in space-based telescopes located far from a planetary magnetic field [13].

In this paper we present a fuzzy logic-based algorithm for converting celestial coordinates into image coordinates for even complex combinations of wide-

angle non-linear optical distortions, and an algorithm for classification of peaks in an astronomical frame and rejection of cosmic ray hits. In Section 2 we present the Night Sky Live sky survey project which is used as a test case for the presented algorithms, in Section 3 we present the transformation formula for converting celestial coordinates to image coordinates and in Section 4 we present the cosmic ray hit rejection algorithm.

2 The *Night Sky Live!* All-sky Survey

The sky has been monitored by astronomers since the very beginning of written history, and probably even earlier. Ancient astronomers kept records of the positions of constellations and bright stars, and noted unusual events such as comets, meteors, and supernovae. The Night Sky Live [12] sky survey is humanity's first attempt to create and sustain a systematic and continuous record of the night sky. Several nodes called *CONCAM*, located at some of the world's premier observatories form together the *Night Sky Live!* network. Each CONCAM includes a CCD camera, a wide-angle fisheye lens and an industrial PC running Linux Red-Hat. The CCD cameras used are SBIG ST-1001E or SBIG ST-8. The lenses are SIGMA F4-EX for the ST-1001E and Nikon FC-E8 for the ST-8 CCDs. The wide-angle lenses allow recording full 2π steradians in one frame, and can detect stars down to visual magnitude 6.8 near the image center. The pictures are 1024×1024 FITS format [23] images, which is a standard format in astronomical imaging. Each CONCAM records a 180-second exposure every 3 minutes and 56 seconds and transmits the images to a main server, where they can be accessed at http://nightskylive.net.

Currently there are 10 *CONCAM* nodes located in Mauna Kea and Haleakala (Hawaii), Cerro Pachon (Chille), Kitt Peak (Arizona), Mt. Wilson (California), Rosemary Hill (Florida), Siding Spring (Australia), Wise Observatory (Israel), Canary Islands and South Africa.

Fig. 1 is a sample Night Sky Live image taken at Mauna Kea observatory. The picture captures the whole night sky so that the big circle is the horizon. The white spots inside the circle are the brightest stars, and the galactic plain (milky way) can also be seen near the center of the image. The structure at the upper left side is one of the telescope domes.

3 Fuzzy Logic-Based Coordinate Transformations

Celestial coordinates can be accurately calculated for any given star or planet. However, in order to associate the sources of light appearing in a given image

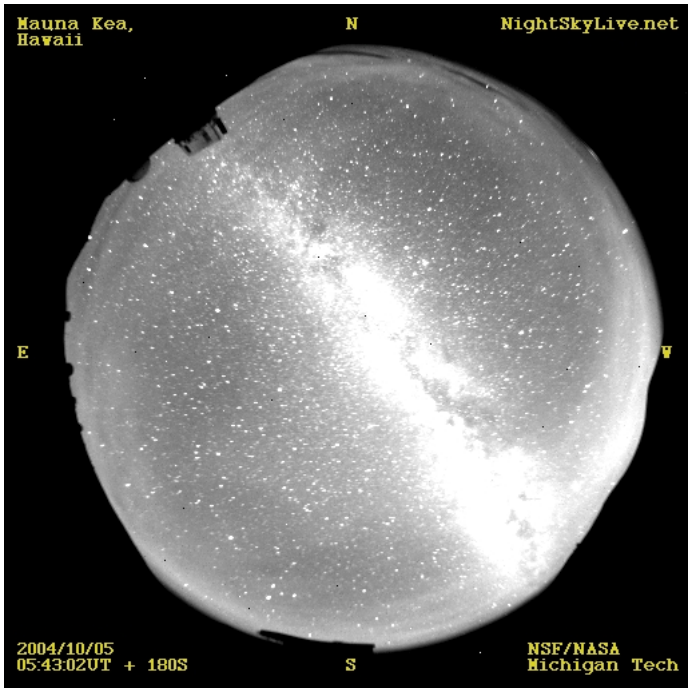

Fig. 1. A Night Sky Live picture taken at Mauna Kea observatory, Hawaii

with known stellar objects, one needs to transform the celestial coordinates of the stellar objects to image (x, y) coordinates, where the light emitted from each object is expected to appear. In an imperfect world of non-linear wide-angle optics, imperfect optics, inaccurately pointed telescopes, and defect-ridden cameras, accurate transformation of celestial coordinates to image coordinates is not always a trivial first step.

3.1 The Transformation Formula

The transformation of celestial topocentric spherical sky coordinates (Azimuth, Altitude) to local Cartesian image coordinates can be defined by a set of two functions:

$$(altitude, azimuth) \longmapsto x \tag{1}$$

$$(altitude, azimuth) \longmapsto y \tag{2}$$

On a CCD image, pixel locations can be specified in either Cartesian or polar coordinates. Let x_{zen} be the x coordinate (in pixels) of the zenith in the image, and y_{zen} be the y coordinate of the zenith. x and y coordinates of any given star visible in an astronomical image can be computed as follows:

$$x = x_{zen} + \sin(\theta) \cdot R \tag{3}$$

$$y = y_{zen} + \cos(\theta) \cdot R \tag{4}$$

Where θ is a polar azimuthal angle and R is a polar radial distance.

In order to use these equations it is necessary to compute a polar distance and angle for given objects. Given the observer's latitude and longitude, this can be done by converting the celestial coordinates (azimuth and altitude) of a given stellar object at a given time to the corresponding angle and distance in the image. Since the azimuth and altitude of any given bright star or planet at any given time can be easily computed, the only missing link here is the transformation of the altitude and azimuth to image coordinates, so the object can be found in the image.

3.2 Reference Stars

Each of the two models is built based on manually identified reference stars. A reference star can be any star within an image that was correctly associated with the corresponding stellar object. Being familiar with the night sky, we can inspect the frame by eye and identify the stellar objects that appear in it. The image (x, y) coordinates of the object can be taken from the peak of its point spread function (PSF). Each reference star provides a record with the following fields: azimuth, altitude, polar azimuthal angle, and polar radial distance.

Each identified star contributes an azimuth and an altitude (by basic astronomy) and also an angle and distance (by measurement from the image). These provide the raw data for constructing a mapping between the two, using the fuzzy logic model that will be described later in the paper. In order to obtain an accurate fuzzy logic model that calculates the angle, it is necessary to select reference stars that uniformly cover the entire image. This assures that the calculation of one value will depend on reference points that are relatively close to it. This is also true for the model that calculates the distance.

3.3 Building the Fuzzy Logic Model

In order to transform celestial coordinates into image coordinates, two different fuzzy logic models are being built based upon the two transformations:

$$f_1 : azimuth \longmapsto angle \tag{5}$$

$$f_2 : altitude, azimuth \longmapsto distance \tag{6}$$

Here, *altitude*, *azimuth* and *angle* are angular measures, while *distance* is measured in pixels. Each transformation (f_1 and f_2) is computed by a different fuzzy logic model, thus one model calculates the angle and the other calculates the distance. The fuzzy logic model f_1 has one antecedent fuzzy variable and one consequent fuzzy variable, while the fuzzy logic model f_2 has two antecedent variables (*altitude*,*azimuth*) and one consequent fuzzy variable (the radial distance).

3.4 Converting Azimuth to Polar Angle on the CCD (f_1)

The first model (f_1) is built according to the reference stars such that each reference star adds to the model one fuzzy set and one fuzzy rule. Each fuzzy set is associated with a membership functions that is built in the form of a triangle [26]. Each of the membership functions reaches its maximum at the reference value, and intersects with the x-axis at the reference values of its neighboring reference stars. For instance, suppose we would like to build the fuzzy logic model with a data set that contains the following four reference stars:

azimuth	**altitude**	**angle**	**distance**
0	ϵ_0	θ_0	R_0
α_1	ϵ_1	θ_1	R_1
α_2	ϵ_2	θ_2	R_2
α_3	ϵ_3	θ_3	R_3

The first reference star maps azimuth 0^o. Assuming $\alpha_1 < \alpha_2 < \alpha_3$, the membership functions that will be added to the model are described in Fig. 2.

Using the simple trapezoidal membership function μ given in Equation 7, the parametric representation of the membership functions is specified in Table 1

$$\mu(x,a,b,c,d) = \begin{cases} 0 & x < a, x > d \\ 1 & b \leq x \leq c \\ \frac{x-a}{b-a} & a \leq x < b \\ 1 - \frac{x-c}{d-c} & c < x \leq d \end{cases} \tag{7}$$

F_0 to F_4 are the membership functions of the fuzzy sets FS_0 to FS_4 that were created by the reference stars (the function F_m is the membership function of the fuzzy set FS_m). The membership functions are built such that almost all azimuth values belong (with non-zero membership) to two fuzzy sets. Only the points of maximum (0,α_1,α_2,α_3,360) have a non-zero membership to just one set. Since the f_1 model is intensively used in the analysis of astronomical images containing 1 to 16 million pixels and thousands of stars, the simple

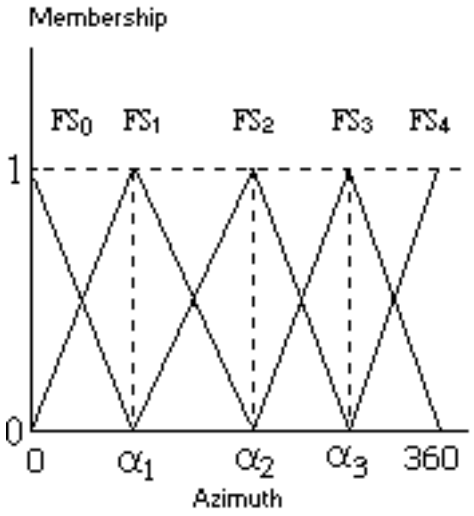

Fig. 2. The membership functions of the fuzzy sets (FS_0 to FS_4) created by the four reference values $(0, \theta_0), (\alpha_1, \theta_1), (\alpha_2, \theta_2), (\alpha_3, \theta_3)$ for $f_1(azimuth \longmapsto angle)$.

Table 1
Parametric representation of the membership functions of the fuzzy sets FS_0 to FS_4

Function	a	b	c	d
F_0	0	0	0	α_1
F_1	0	α_1	α_1	α_2
F_2	α_1	α_2	α_2	α_3
F_3	α_2	α_3	α_3	360
F_4	α_3	360	360	360

triangular membership functions are used here for their low computational cost.

The reasoning procedure used in this model is based on Takagi-Sugeno, which is efficient when the fuzzy logic model is built according to a set of singleton values [20, 21]. Since *weighted average* is used, the consequent part of each rule is a function providing an immediate crisp value. Therefore, the fuzzy rules that will be added to the model are:

If x is $FS_0 \longmapsto \theta_0 + \frac{x-0}{\alpha_1-0}(\theta_1 - \theta_0)$

If x is $FS_1 \longmapsto \begin{cases} \theta_1 + \frac{x-\alpha_1}{\alpha_2-\alpha_1}(\theta_2 - \theta_1) & x > \alpha_1 \\ \theta_1 - (1 - \frac{\alpha_1-x}{\alpha_1-0})(\theta_1 - \theta_0) & x \leq \alpha_1 \end{cases}$

$$\text{If x is } FS_2 \longmapsto \begin{cases} \theta_2 + \frac{x-\alpha_2}{\alpha_3-\alpha_2}(\theta_3-\theta_2) & x > \alpha_2 \\ \theta_2 - (1 - \frac{\alpha_2-x}{\alpha_2-\alpha_1})(\theta_2-\theta_1) & x \leq \alpha_2 \end{cases}$$

$$\text{If x is } FS_3 \longmapsto \begin{cases} \theta_3 + \frac{x-\alpha_3}{360-\alpha_3}(\theta_4-\theta_3) & x > \alpha_3 \\ \theta_3 - (1 - \frac{\alpha_3-x}{\alpha_3-\alpha_2})(\theta_3-\theta_2) & x \leq \alpha_3 \end{cases}$$

$$\text{If x is } FS_4 \longmapsto 360 - (1 - \tfrac{360-x}{360-\alpha_3})(360-\theta_3)$$

For instance, suppose that the first three reference stars have azimuths of 0^o, 10^o, and 20^o, and their polar angles are 3^o, 12^o, and 22^o respectively such that $\alpha_0 = 0^o$, $\theta_0 = 3^o$, $\alpha_1 = 10^o$, $\theta_1 = 12^o$, $\alpha_2 = 20^o$, and $\theta_2 = 22^o$. Each reference star adds to the model one fuzzy such that the membership functions of the first two fuzzy sets FS_0, FS_1 are:

$$F_0(x) = \begin{cases} 1 - \frac{1}{10} \cdot x & 0 \leq x \leq 10 \\ 0 & x < 0 \text{ or } x > 10 \end{cases}$$

$$F_1(x) = \begin{cases} \frac{1}{10} \cdot x & 0 \leq x < 10 \\ 1 - \frac{1}{10} \cdot (x-10) & 10 \leq x \leq 20 \\ 0 & x < 0 \text{ or } x > 20 \end{cases}$$

Each reference star also adds one fuzzy rule so that the first two fuzzy rules are:

$$\text{If x is } FS_0 \longmapsto 3 + \tfrac{x-0}{10}(12-3)$$

$$\text{If x is } FS_1 \longmapsto \begin{cases} 12 + \frac{x-10}{10}(22-12) & x > 10 \\ 12 - (1 - \frac{10-x}{10})(12-3) & x \leq 10 \end{cases}$$

3.5 Converting Altitude and Azimuth to Radial Distance on the CCD (f_2)

Unlike the simpler f_1 model used for transforming the azimuth to angle, the computation of the distance (in pixels) from (x_{zen}, y_{zen}) should be computed based on *two* parameters, which are the altitude and the azimuth. Using both the altitude and azimuth allows the model to deal with asymmetric behavior of the optics as well as inaccurate orientational information. In other words, using the assumption that the optics orientation is directly at the zenith and the distortion of the optics and hardware is completely symmetric, a reference point at a certain azimuth would allow calculating the distance of a point at the same altitude but at a different azimuth. However, this is not always the case. For instance, a stellar object with the azimuth of 0^o (north) and altitude of 30^o can be at distance of 150 pixels from (x_{zen}, y_{zen}), while another stellar object at the same altitude (30^o) but at azimuth of 60^o will be at distance of 155

pixels from (x_{zen}, y_{zen}). Moreover, (x_{zen}, y_{zen}) does not necessarily appear in the center of the frame, and the frame is not necessarily centralized. Therefore, when computing the distance of a stellar object from (x_{zen}, y_{zen}), it is required to take not only the altitude of the stellar object into considerations, but also the azimuth. In order to do that, the fuzzy logic model that calculates the distance is built according to reference stars that not only have different altitudes, but also different azimuths.

We build the f_2 model that computes the distance using four different sets of reference stars such that each set contains reference stars that share approximately the same azimuth. For the sake of simplicity, the first set contains reference stars that are near azimuth 0^o, the second set contains reference stars near azimuth 90^o, and the other two sets contain stars near azimuth of 180^o and 270^o respectively. I.e., all reference stars used for this model should be fairly close to the azimuth of 0^o, 90^o, 180^o or 270^o. In order to use the four sets of reference stars, four new fuzzy sets are added to the model. Those fuzzy sets are "North","East", "South", and "West". The membership functions of the fuzzy sets are described in Fig. 3.

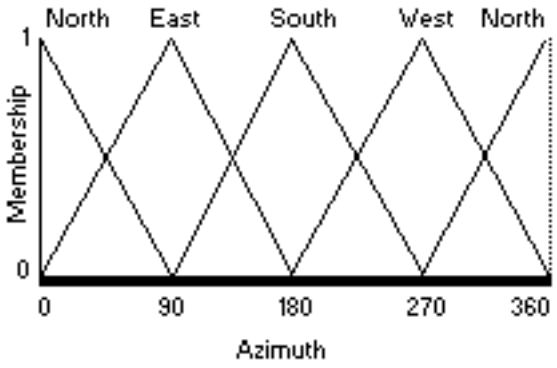

Fig. 3. The fuzzy sets of the four directions (North, South, East, West).

Building f_2 model can be demonstrated by an example: Suppose the model is built based on the following 6 reference stars listed in Table 2.

As with f_1, each reference star adds to the model one fuzzy set that has a triangle membership function. For instance, the membership function of the fuzzy set added by S_1 reaches its maximum of unity at 68, and intersects with the x-axis at the points of maximum of its neighboring reference stars. The neighboring stars are the two stars that their altitudes are closest to the altitude of S_1 (such that one is greater than 68^o and one is smaller than 68^o), *and* have approximately the same azimuth as S_1. In this example, one neighboring star would be S_2 and the other would be S_3. All three stars share approximately the same azimuth (North). Therefore, the fuzzy set *Alt68N* added by S_1 will have a triangular membership function that reaches its maximum at 68, and intersects with the x-axis at 62 and 72. This membership function is

Table 2
Reference stars

star	azimuth	altitude	angle	distance
S_1	0^o	68^o	2.4^o	215 pixels
S_2	359^o	62^o	1.2^o	224 pixels
S_3	1.5^o	72^o	3.4^o	206 pixels
S_4	90^o	66^o	94.2	180 pixels
S_5	91^o	62^o	95.6	188 pixels
S_6	91.6^o	70^o	95.9	172 pixels

described in Fig. 4.

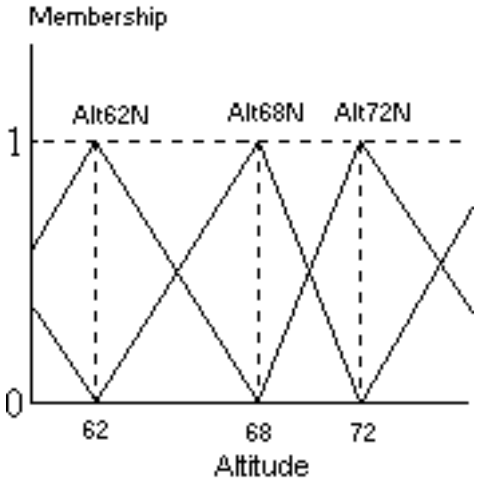

Fig. 4. The membership function of the fuzzy set *Alt68N*. The figure also include parts of the membership functions of *Alt62N* and *Alt72N* that were added by the two neighboring reference stars S_2 and S_3.

The membership function added by S_1 is:

$$F_{Alt68N}(x) = \begin{cases} \frac{x-62}{68-62} & 62 < x \leq 68 \\ 1 - \frac{x-68}{72-68} & 68 < x \leq 72 \\ 0 & x < 62 \text{ or } x > 72 \end{cases}$$

Since the azimuth of S_1 is approximately north, the fuzzy rule added to the model by S_1 is:

If Alt is Alt68N $\wedge$ Az is North $\longmapsto$ $\frac{F_1(Alt)\cdot(90-Az)+F_2(Alt)\cdot Az}{90}$
Where
$F_1 = \begin{cases} 215 - \frac{x-68}{72-68}(215-206) & Alt > 68 \\ 215 + \frac{68-x}{68-62}(224-215) & Alt \leq 68 \end{cases}$
and

$$F_2 = \begin{cases} 180 - \frac{x-66}{70-66}(180-172) & Alt > 66 \\ 180 + \frac{66-x}{66-62}(188-180) & Alt \leq 66 \end{cases}$$

Intuitively, This rule is more significant for stars that appear in the northern part of the sky (and in this case, also at an altitude of around 68^o). The practical effect of this rule will be stronger as the coordinates are closer to the 68^o parallel.

3.6 Example Application to Night Sky Live Data

The fuzzy logic based transformation formula is used by NSL for converting the celestial coordinates to image coordinates, so known catalogued stellar objects can be associated with PSFs that appear in the Night Sky Live frames. One simple task that is enabled by this transformation formula is the annotation of the all-sky images with the names of bright stars, constellations and planets. This task is mostly used for educational or "cosmetic" purposes. A more important task is the automatic detection of non-catalogued objects. This task is required for automatic detection of meteors, comets, novae and supernovae, as well as other astronomical phenomena visible in the night sky. The following algorithm uses the transformation formula in order to associate PSFs in the image to stellar objects.

```
1. function check_stars(image, date_time)
2. image_PSFs ← GetPSFs(image)
3. for each image_PSF_cords in image_PSFs do
4. begin
5.    min_distance ← ∞
6.    for each star in catalog do
7.       star_celestial_cords ← CelestialCoordinates(star, date_time)
8.       if InView(star_celestial_cords) then
9.          image_cords ← AltAz2XY(star_celestial_cords)
10.          if distance(image_cords,image_PSF_cords) < min_distance then
11.             min_distance ← distance(image_cords, image_PSF_cords)
12.      end if
13.   end for
14.   if min_distance < TOLERANCE then
15.   associate(image_PSF_cords,star)
16. end for
```

The function *AltAz2XY* transforms celestial coordinates to image coordinates based on the fuzzy logic models. The function *GetPSFs* returns a list of coordinates of the PSF peaks that appear in the picture. This function can be

implemented by using some available algorithms for detection of sources from astronomical images such as *SExtractor* [3]. The function *InView* returns *true* if its argument coordinates are inside the relevant view of the optical device. In the inner loop the algorithm searches the catalog for a star that should appear closest to the center of the PSF. Since the hardware used for the Night Sky Live project currently cannot get deeper than magnitude 6.8, the catalog being used is a subset of Hipparcos catalog [7] that is sure to include objects this bright. In line 14, the minimum distance found in the inner loop is compared with a constant value *TOLERANCE* that is a tolerance value. Only if the distance is smaller than *TOLERANCE* then *image_cords* and *image_PSF_cords* are considered as referring to the same star.

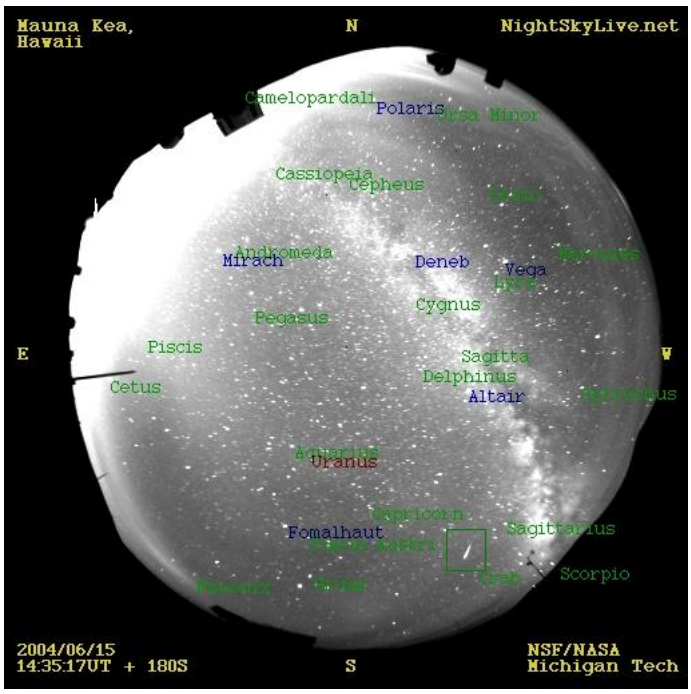

Fig. 5. A Night Sky Live picture with labeled stars, constellations, a planet and a meteor (boxed) processed using the fuzzy logic based transformation formula.

In Fig. 5, taken by the Mauna Kea CONCAM, the bright stars and constellations were annotated automatically using the above transformation. The coordinates of one bright point source (appears at the lower right of the frame) could not be associated with any catalogued object so it was automatically marked with a yellow square. This object is believed to be a meteor.

The present fuzzy-logic algorithms allow practically 100 percent chance of accurate identification for NSL stars down to a magnitude of 5.6. We are currently unaware of any exceptions. A previously used NSL identification algorithm that employed a straightforward analytic transformation was only accurate to about magnitude 3.5, although that was somewhat dependent on the NSL station.

Another useful product of the newly accurate identifications is the automatic generation of photometry files. The ability to associate each PSF with a cataloged star allows the system to provide continuous monitoring of many bright stars. This information is provided in the form of text (XML tagged) files. Each frame produces one text file that lists all PSFs that were detected in the frame and the name and catalogue number of the stellar object associated with it. It also lists some additional data about each detected object such as the previously cataloged visual magnitude, spectral type, and celestial coordinates. Identification allows other algorithms to process the frame and match each star with real time photometric data such as estimated counts of the background and the counts of the PSF.

The inverse computation of the presented transformation formula (converting angle and distance to altitude and azimuth) can be built in the same method described in this paper, with the exception of using the angle and distance for defining the membership functions, while the azimuth and altitude are used as the crisp output values of the fuzzy rules. This transformation formula is currently being used by NSL for computing 3D trajectories of meteors detected by the twin CONCAMs located at Mauna Kea and Haleakala.

4 A Fuzzy Logic Based Algorithm for Cosmic Ray Hit Rejection

The presence of cosmic ray hits in astronomical CCD frames is frequently considered as a disturbing effect. Cosmic rays add an undesirable signal to astronomical images, and can weaken algorithms for astronomical image processing. For instance, it can decrease the compression factor of astronomical image compression algorithms [13], and can disturb the operation of autonomous astronomical pipelines [1, 2, 14]. Exposures taken at high altitude observatories get more cosmic ray hits than sea level observatories. This becomes even more significant in space-based telescopes located far from a planetary magnetic field [13].

Several methods for cosmic ray hit rejection have been proposed. One common technique is by comparing several exposures of the same field [9, 18, 24]. However, exposures of the same fields are not always available. Other approaches, such as [15, 6, 16, 17], have been proposed in order to perform cosmic ray hit rejection in a single CCD exposure. These approaches include Laplacian edge detection [6], artificial neural network [17], analysis of the histogram of the image data [15] and filtering by adapted point spread functions [16]. The more difficult cases are when some of the multiple-pixel cosmic ray hits are larger than some of the point spread functions of true astronomical sources [6]. Thus, reasonably trained humans can usually perform this task with a considerable percentage of accuracy. In this section we present an algorithm that aims to

reject cosmic ray hits based on human perception.

4.1 Manual Detection of Cosmic Ray Hits

Cosmic ray hits in astronomical exposures are usually noticeably different then point spread functions of true astronomical sources, and a reasonably trained human can usually tell between the two. One examining an astronomical frame can notice that cosmic ray hits are usually smaller than PSFs of astronomical sources, and their edges are usually sharper. Although many cosmic ray hits are not larger than just one pixel, in some cases they can be larger than some of the point spread functions of astronomical sources [6]. An observer trying to manually detect cosmic ray hits in an astronomical frame would probably examine the edges and the surface size of the peaks. For instance, if the surface size of the peak is very small and it has sharp edges, it would be classified as a cosmic ray hit. If the surface size of the peak is larger and its edges are not very sharp, it would be probably classified as a PSF of an astronomical source. Since some of the cosmic ray hits have only one or two sharp edges, it is also necessary to examine the sharpest edge of the PSF.

This intuition can be summarized by a set of intuitive natural language rules such as:

(1) If the peak is small and the edges are sharp then the peak is a cosmic ray hit.
(2) If the peak is large and the edges are not very sharp then the peak is not a cosmic ray hit.
(3) If the peak is medium and most of the edges are not very sharp except from one extremely sharp edge then the peak is a cosmic ray hit.
(4) If the peak is small and the edges are moderately sharp then the peak is a cosmic ray hit.
(5) If the peak is large and the edges are not sharp except from one edge that is moderately sharp then the peak is not a cosmic ray hit.

4.2 A Human Perception-Based Fuzzy Logic Model

The first step in compiling the rules of intuition described above into a fuzzy logic model is to define the antecedent and consequent fuzzy variables. The antecedent variables in this model are the surface size of the peak (in pixels), the sharpness of the sharpest edge (in σ, where σ is the estimated noise) and the average sharpness of the edges (also in σ). The consequent variable is the classification of the peak. The domain of this variable is {Yes/1,No/0}, such that YES/1 means that the peak is classified as a cosmic ray hit and NO/0

means that the peak is classified as a true astronomical source. Therefore, the model can be defined by the following function *f*:

f: *surface_size*, *sharpest_edge*, *average_edge* $\longmapsto$ {0,1}

The surface size is determined by counting the pixels around the peak that are at least 3σ brighter than the local background. When a pixel less than 3σ above the local background is reached, the pixel is not counted, and the edge sharpness is determined as the difference between the value of that pixel and the value of its neighboring pixel (in the direction of the peak).

Since the shapes of cosmic ray hits are dependent on the gain and pixel size of the CCD chip, different chips require different membership functions. In the example provided in this paper we assume a CCD with pixel size of 24μm and gain of one electron, but the membership functions can be easily adjusted to other CCD chips.

The fuzzy sets defined on the surface size of the peak are described in Fig. 6.

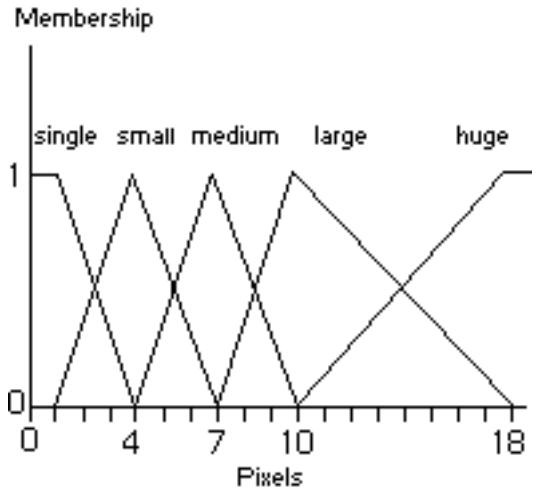

Fig. 6. Fuzzy sets of the surface size: single, small, medium, large, huge.

Using the simple trapezoidal membership function μ given in Equation 7, the parametric representation of the membership functions is given in Table 3

The antecedent variables *sharpest_edge* and *average_edge* use the same fuzzy sets. The fuzzy sets defined on these variables are *low*, *moderate*, *sharp* and *extreme*, as described in Fig. 7

Using the simple trapezoidal membership function μ given in Equation 7, the parametric representation of the membership functions described in Fig. 7 is specified in Table 4

Table 3
Parametric representation of the membership functions of surface size

function	a	b	c	d
F_{single}	0	0	1	4
F_{small}	1	4	4	7
F_{medium}	4	7	7	10
F_{large}	7	10	10	18
F_{huge}	10	18	∞	∞

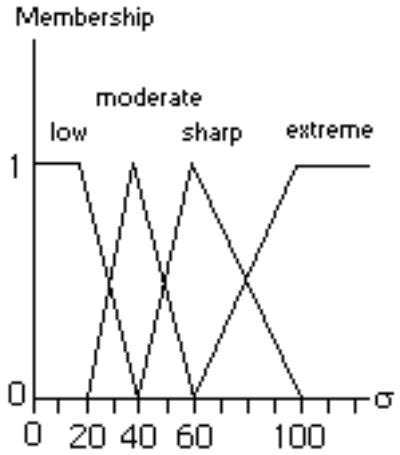

Fig. 7. Fuzzy sets of the edge sharpness

Table 4
Parametric representation of the membership functions of edge size

Function	a	b	c	d
F_{single}	0	0	20	40
F_{small}	20	40	40	60
F_{medium}	40	60	60	100
F_{large}	60	100	∞	∞

As will be explained in Sect. 4.3, the computation has to be applied to each pixel in the image. Since astronomical images typically contain 1 to 16 million pixels, the simple triangular membership functions are used for their low computational cost.

The fuzzy rules are defined using the membership functions of the antecedent variables and the domain of the consequent variable {0,1}, and are based on the natural language rules of intuition described in Sect. 4.1. The antecedent parts of the fuzzy rules with the consequent part *1* (cosmic ray hits) are:

single, low, moderate	single, low, sharp	single, low, extreme
single, moderate, low	single, moderate, moderate	single, moderate, sharp
single, moderate, extreme	single, sharp, low	single, sharp, moderate
single, sharp, sharp	single, sharp, extreme	single, extreme, low
single, extreme, moderate	single, extreme, sharp	single, extreme, extreme
small, low, sharp	small, low, extreme	small, moderate, low
small, moderate, moderate	small, moderate, sharp	small, moderate, extreme
small, sharp, low	small, sharp, moderate	small, sharp, sharp
small, sharp, extreme	small, extreme, low	small, extreme, moderate
small, extreme, sharp	small, extreme, extreme	medium, moderate, extreme
medium, sharp, low	medium, sharp, moderate	medium, sharp, sharp
medium, sharp, extreme	medium, extreme, low	medium, extreme, moderate
medium, extreme, sharp	medium, extreme, extreme	large, sharp, sharp
large, sharp, extreme	large, extreme, moderate	large, extreme, sharp
large, extreme, extreme		

The fuzzy rules with consequent part *0* (not cosmic ray hits) are:

single, low, low	small, low, low	small, low, moderate
medium, low, low	medium, low, moderate	medium, low, sharp
medium, low, extreme	medium, moderate, low	medium, moderate, moderate
medium, moderate, sharp	large, low, low	large, low, moderate
large, low, sharp	large, low, extreme	large, moderate, low
large, moderate, moderate	large, moderate, sharp	large, moderate, extreme
large, sharp, low	large, sharp, moderate	large, extreme, low
huge, low, low	huge, low, moderate	huge, low, sharp
huge, low, extreme	huge, moderate, low	huge, moderate, moderate
huge, moderate, sharp	huge, moderate, extreme	huge, sharp, low
huge, sharp, moderate	huge, sharp, sharp	huge, sharp, extreme
huge, extreme, low	huge, extreme, moderate	huge, extreme, sharp
huge, extreme, extreme		

The computation process is based on zero-order Takagi-Sugeno [20, 21]. Normally, when the input values are typical to cosmic ray hits the value of the consequent fuzzy variable is close to 1, while non-cosmic ray hits provide values closer to 0. The threshold was set to 0.5, so that if the value of the consequent variable is greater than 0.5 the PSF is classified as a cosmic ray hit. Otherwise, the PSF is classified as non-cosmic ray hits.

Although data driven approaches are commonly used for two-class classification problems, the described knowledge-driven approach was found effective and has some advantages over data-driven models: First, the number of samples required for building the knowledge-driven model is much smaller than in data-driven approaches. The manual data classification and collection of cosmic-ray hits of different shapes and sizes can become an exhausting task, and may introduce a barrier to astronomers who wish to implement the model. Another advantage is that the knowledge-driven model can be easily migrated to other instruments with different CCD chips, without the necessity to recollect the data. For instance, if the pixel size is $0.14\mu m$ rather than $0.28\mu m$ (meaning that the area of each pixel is $\frac{1}{4}$ of the original area), the points of maximum unity of the membership functions of the size should be simply multiplied by four, such that if the maximum unity of the membership function of *small* is 4, then in the new model the maximum unity of *small* will be 16.

4.3 Using the Fuzzy Logic Model

The fuzzy logic model is used in order to classify peaks in the frame as cosmic ray hits or non-cosmic ray hits. Each cosmic ray hit or PSF has one (or several) brightest pixels that can be considered as the center of the peak. In the presented algorithm, searching for peaks in a FITS frame is performed by comparing the value of each pixel with the values of its 8 neighboring pixels. If the pixel is equal or brighter than *all* its neighboring pixels, it is considered as a center of a peak. After finding the peaks in the frame, the fuzzy logic model is applied in order to classify the peaks as cosmic ray hits or non-cosmic ray hits.

If a background pixel happens to be brighter or equal to its 8 neighboring pixels, it will be mistakenly considered as a center of a peak. The probability of a this event is $0.5^8 = 0.00390625$. For instance, in an astronomical frame of 1024×1024, 4096 background pixels are expected to be mistakenly considered as peaks. However, since the computation process for each peak is relatively fast, these additional peaks do not significantly slow down the algorithm.

4.4 Performance of the Algorithm

The proposed algorithm have been implemented and used by the *Night Sky Live* project. Measurements of the performance of the algorithm were taken using 24 *Night Sky Live* all-sky exposures. Each NSL frame contains an average of 6 noticeable cosmic ray hits brighter than 20σ, and around 1400 astronomical sources brighter than 20σ above their local background. Out of 158 cosmic ray hits that were tested, the algorithm did not reject 4, and mistakenly rejected 6 true astronomical sources out of a total of 31,251 PSFs.

In order to compare the performance of the proposed algorithm to previously reported algorithms, we used the data and comparison approach proposed by [8]. The results are given in Table 5, and show that the proposed algorithm is favorably comparable in terms of false positives. The fuzzy logic-based algorithm also has a clear advantage in terms of computational complexity. While some proposed cosmic ray hit rejection algorithms are relatively slow [16, 6, 15] (full processing of an image is a matter of several minutes), the presented algorithm can process a 1024×1024 integer FITS frame in less than 4 seconds, using a system equipped with an Intel Pentium IV 2.66 MHZ processor and 512 MB of RAM.

Algorithm	Cosmic-Rays Rejected	False Positives (pixels)
Van Dokkum [6]	86%	1.5%
Rhoads [16]	78%	1.4%
Pych [15]	76%	1.4%
The Proposed Algorithm	84%	0.6%

Table 5
Comparison of the accuracy of Van Dokkum, Rhoads, Pych and the proposed algorithm

5 Conclusion

The emerging field of robotic telescopes and autonomous sky surveys introduces a wide range of problems that require complex decision making. In this paper we presented fuzzy logic-based solutions to two basic problems in the field of astronomical pipeline processing, which are star recognition and cosmic ray hit rejection. We showed that fuzzy logic modeling provides the infrastructure for complex decision making required for automatic analysis of astronomical frames, yet complies with the practical algorithmic complexity constraints introduced by the huge amounts of data generated by the astronomical pipelines.

References

[1] T. Axelrod, A. Connolly, Z. Ivezic, J. Kantor, R. Lupton, R. Plante, C. Stubbs, D. Wittman, The LSST data processing pipeline, AAS Meeting 205 (2004) 108.11.

[2] A.C. Becker, A. Rest, G. Miknaitis, R.C. Smith, C. Stubbs, LSST : Image Subtraction & Transient detection techniques, AAS Meeting 205 (2004) 108.12.

[3] E. Bertin, S. Arnouts, SExtractor: Software for source extraction, Astronomy & Astrophysics Supplement 117 (1996) 393–404.

[4] W.J. Borucki, D. Koch, G. Basri, T. Brown, D. Caldwell, E. Devore, E. Dunham, T. Gautier, J. Geary, R. Gilliland, A. Gould, S. Howell, J. Jenkins, Kepler Mission: a mission to find Earth-size planets in the habitable zone, Proc. of the Conf. on Towards Other Earths (2003) 69–81.

[5] M. Clampin, C. Atkinson, L. Feinberg, W. Hayden, P. Lightsey, M. Mountain, S. Texter, Status of the JWST Observatory Design, American Astronomical Society Meeting 203 (2003) 121.01.

[6] P.G. Van Dokkum, Cosmic-ray rejection by Laplacian edge detection, Publications of the Astronomical Society of the Pacific, 113 (2001) 1420–1427.

[7] ESA, The Hipparcos and Tycho catalogues, ESA SP-1200 (1997).

[8] Farage, C.L., Pimbblet, K.A., Evaluation of Cosmic Ray Rejection Algorithms on Single-Shot Exposures, Publications of the Astronomical Society of Australia, 22(3), (2005) 249–256.

[9] D.J. Fixsen, J.D. Offenberg, R.J. Hanisch, J.C. Mather, M.A. Nieto-Santisteban, R. Sengupta, H. S. Stockman, Cosmic-ray rejection and readout efficiency for large-area arrays, Publications of the Astronomical Society of the Pacific 112 (2000) 1350.

[10] J. Loveday, The Sloan digital sky survey (SDSS), Journal of the British Astronomical Association, 113(6) (2003) 364.

[11] N. Kaiser, Pan-STARRS: a wide-field optical survey telescope array, Proceedings of the SPIE 5489 (2004) 11–22.

[12] R.J. Nemiroff, J.B. Rafert, Toward a continuous record of the sky, The Publications of the Astronomical Society of the Pacific 111 (1999) 886–897.

[13] J.D. Offenberg, R. Sengupta, D.J. Fixsen, H.S. Stockman, M.A. Nieto-Santisteban, S. Stallcup, R.J. Hanisch, J.C. Mather, Cosmic ray rejection with NGST, in ASP Conf. Ser. 172 (1999) 141.

[14] S. Otuairisg, A. Golden, R.F. Butler, A. Shearer, B. Voisin, A pipeline for automatically processing and analyzing archival images from multiple instruments, Ground-based Telescopes 5493 (2004) 467–473.

[15] W. Pych, A Fast Algorithm for cosmic-ray removal from single images, Publications of the Astronomical Society of the Pacific 116 (2004) 148–153.

[16] J.E. Rhoads, Cosmic-ray rejection by linear filtering of single images, Publications of the Astronomical Society of the Pacific 112 (2000) 703–710.

[17] S. Salzberg, R. Chandar, H. Ford, S.K. Murphy, R. White, Decision trees for automated identification of cosmic-ray hits in Hubble Space Telescope

images, Publications of the Astronomical Society of the Pacific 107 (1995) 279–288.

[18] R.A. Shaw, K. Horne, Noise model-based cosmic ray rejection for WF/PC images, ASP Conf. Ser. 25, Astronomical Data Analysis - Software and Systems (1992) 311.

[19] J.A. Smith, S.S. Allam, R.C. Bohlin, S.E. Duestua, D. Ebbets, S. Kent, M. Lampton, B.E. Laubscher, N. Mostek, The SNAP Standard Star Program, American Astronomical Society Meeting 203 (2003) 82.22.

[20] T. Takagi, M. Sugeno, Derivation of fuzzy Control rules from human Operator's control actions, Proc. of the IFAC Symp. on Fuzzy Information, Knowledge Representation and Decision analysis (1983) 55–60.

[21] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, IEEE Trans. Syst. Man & Cybern. 20(2) (1985) 116–132.

[22] J.A. Tyson, Survey and other telescope technologies and discoveries, Proceedings of the SPIE 4836 (2002) 10–20.

[23] D.C. Wells, E.W. Greisen, R.H. Harten, FITS - a flexible image transport system, Astronomy and Astrophysics Supp. Ser. 44, (1981) 363.

[24] R.A. Windhorst, B.E. Franklin, L.W. Neuschaefer, Removing cosmic-ray hits from multiorbit HST wide field camera images, Publications of the Astronomical Society of the Pacific 106, (1994) 798–806.

[25] D.G. York, The Sloan digital sky survey: Technical Summary, The Astronomical Journal 120, (2000) 1579–1587.

[26] L.A. Zadeh, Fuzzy sets, Information and Control 8 (1965) 338–353.