Comparative analysis of methods for cloud segmentation in ground-based infrared images

Guillermo Terrén-Serrano [*], Manel Martínez-Ramón

Department of Electrical and Computer Engineering, The University of New Mexico, Albuquerque, NM, 87131, United States

ARTICLE INFO

ABSTRACT

The increasing penetration of photovoltaic systems in the power grid makes it vulnerable to cloud shadow projection. Real-time cloud segmentation in ground-based infrared images is important to reduce the noise in intra-hour global solar irradiance forecasting. We present a comparison between discriminative and generative models for cloud segmentation. The performances of supervised and unsupervised learning methods in cloud segmentation are evaluated. The discriminative models are solved in the primal formulation to make them feasible in real-time applications. The performances are compared using the j-statistic. Infrared image preprocessing to remove stationary artifacts increases the overall performance in the analyzed methods. The inclusion of features from neighboring pixels in the feature vectors leads to a performance improvement in some of the cases. Markov Random Fields achieve the best performance in both unsupervised and supervised generative models. Discriminative models solved in the primal yield a dramatically lower computing time along with high performance in the segmentation. Generative and discriminative models are comparable when preprocessing is applied to the infrared images.

© 2021 Published by Elsevier Ltd.

1. Introduction

A large power grid system fully operated using only renewable power in Europe can be theoretically possible by 2050 [1], and the projected solar energy share will increase to 47% by 2050 in the USA [2]. Clouds increase or decrease the global solar irradiance reaching Earth's surface. This is of great importance when a considerable percentage of the energy in a power grid is generated using large Photovoltaic (PV) systems [3]. Even when the PV arrays in a power plant are arranged in a configuration capable of attenuating the effects caused by moving clouds, cloud shadows produce interruptions in energy generation which may be out of the grid operator's admissible range [4]. Moving clouds have an effect not only on the generation of energy from PV systems, but also on solar thermal power plants [5]. The inclusion of cloudiness information from sky images into a statistical model for forecasting Global Solar Irradiance (GSI) improves the overall performance of the prediction [6].

Computer recognition of clouds is a geospatial information problem [7]. The tropopause limits the range of cloud formations, which seasonally varies across latitudes [8]. Different cloud types are expected to be found at a different range of altitudes [9]. When using features extracted from color intensity channels, cloud patterns inferred from data acquiesced at different latitudes may not be correlated. Feature extraction methods based on Gabor filter texture analysis and statistics are more easily replicable across databases [10].

In GSI forecasting [11], ground-based methods without features extracted from clouds are not effective in intra-hour forecasting and are generally used when the forecasting horizon is hours-ahead [12—17]. For an intra-day hours-ahead forecast, the effects of clouds on ground-level solar irradiance can be assessed using satellite images [18,19] and the accuracy of numerical weather models for GSI nowcasting is improved when cloudiness information is extracted from satellite images [20]. Ground measurements (GSI or PV power) are an option for intra-day forecasting, but are not capable of predicting ramp events (when a cloud will abruptly affect a PV system [21]). Ground-based sky imaging is the most suitable method in applications requiring intra-hour GSI forecasting [22—24].

When using visible light ground-based sensors, the circumsolar region appears saturated in images including the Sun [25,26].

---

* Corresponding author.
E-mail address: guille.terren@gmail.com (G. Terrén-Serrano).

Abbreviations

CNN	Convolutional neural networks
DAQ	Data acquisition system
EM	Expectation-maximization
FOV	Field of view
GDA	Gaussian discriminant analysis
GMM	Gaussian mixture model
GPC	Gaussian process for classification
GSI	Global solar irradiance
ICM	Iterated conditional modes
IR	Infrared
LOO	Leave-one-out
MALR	Moist adiabatic lapse rate
MAP	Maximum a posteriori
ML	Maximum likelihood
MRF	Markov random field
NBC	Naive Bayes classifier
PV	Photovoltaic
ROC	Receiver operating characteristic
RRC	Ridge regression for classification
SA	Simulated annealing
SVC	Support vector classifier
VSH	Voting scheme
WLK	Weighted Lucas-Kanade

Structures that block the Sun's direct irradiance partially obstruct the images [27–30], creating forecasting problems [31]. Nevertheless, total sky imagery [32,33], and fisheye lenses are capable of recording a large Field of View (FOV) [34]. When these technologies are applied with the aim of motion estimation, fisheye lens' distortion should be removed [35]. Recent ground-based Infrared (IR) sky imaging developments increase the FOV of thermal images [36,37]. IR images allow for the derivation of physical features of the clouds such as temperature [38] and height, which are more interpretable for modelling physical processes. Ground-based IR cameras providing radiometric measures [39,40], are used to study statistical cloud features [41], and in earth-space communications applications [42]. Further research shows how to stabilize the thermal image from microbolo meters in atmospheric measurements [43].

Previous investigations in cloud segmentation concluded that pixel segmentation using features extracted from neighboring pixels improves performance [44,45]. Graph models based on neighboring pixels' classification are referred to as MRFs. They are a generalization of the Ising Model, first introduced in ferromagnetic problems [46], and later applied to 2-dimensional crystal lattice problems [47]. The Iterated Conditional Modes (ICM) algorithm, developed for unsupervised training of MRFs in image processing [48], was implemented for IR satellite image cloud segmentation [49], and visible light ground-based images [50].

The superpixel approach speeds-up computing time, but produces a coarse segmentation [51]. Real-time cloud segmentation is a problem for kernel learning methods, as the Gram matrix is generally dense [52,53]. One alternative is the use of primal formulation optimization. The same problem appears with Convolutional Neural Networks (CNN). The required computing time is high [10], although it is considerably reduced when using GPUs [54,55]. Nevertheless, these methods require data augmentation and regularization techniques to avoid overfitting. Otherwise, the conclusions obtained are not comparable between different databases of cloud images, since the distribution of the features will vary. We prove that when effective preprocessing is applied to the IR images to extract informative physical features, discriminative models are faster and have similar accuracy to generative, kernel or CNN methods.

This research utilizes data acquired from an innovative radiometric long-wave IR sky-imaging system, rather than a visible light sky-imaging system. The Data Acquisition (DAQ) system is mounted on a solar tracker. An advantage of using the IR sky-imaging system is that the saturated circumsolar region of the image is smaller. The saturation of the circumsolar area removes necessary information about the clouds for intra-hour GSI forecasting.

Another advantage of the IR imaging system is that thermal images allow for the extraction of physical features that are useful for cloud segmentation. A novelty of this work is the implementation of a preprocessing algorithm to increase the cloud segmentation performances in IR sky images. The proposed preprocessing algorithm applies two models to the IR images [56]. The first model reproduces the scattering effect caused by debris (e.g. water stains and dust) on the outdoor germanium window of the camera. The second model reproduces the effect of direct irradiation from the Sun and scatter irradiation from the atmosphere to remove saturation in the circumsolar region, making it possible to differentiate between the Sun and clouds.

Cloud segmentation is useful to identify which pixels in an image are cloudy and which are clear-sky. This information can then be used in a solar forecasting algorithm. This research contributes to the field of cloud segmentation and solar forecasting through a comparative analysis of generative and discriminative models. The objective is to determine which model performs better in an IR sky-imaging system mounted on a solar tracker. The discriminative methods used in the analysis are: Ridge Regression for Classification (RRC) [57], Support Vector Classifier (SVC) [58] and Gaussian Processes [59] for Classification (GPC). The training and testing time is drastically reduced when the RRC, SVC and GPC models are implemented in their primal formulation, because the number of dimensions obtained after mapping data to the Hilbert space is much smaller compared to the dual formulation. MRFs are part of the analyzed generative models. MRF models are computationally expensive but suitable for segmentation problems, because information from the classification of neighboring pixels is included in the prior. The generative models include effective methods with low computational requirements. The training and testing computation time is improved by simplifying the covariance matrix. The Naive Bayes Classifier (NBC) and k-means clustering are simplifications of the Gaussian Discriminant Analysis (GDA) and Gaussian Mixture Model (GMM) respectively. The performances of generative models are compared between supervised (NBC, GDA and MRF) and unsupervised learning algorithms (k-means, GMM and ICM-MRF). Unsupervised learning models are less time intensive because they do not require labels to train a segmentation model, simplifying training for the user. The Simulated Anneling (SA) algorithm is implemented to perform an intelligent optimization that reduces the testing time of the MRF and ICM-MRF. A voting scheme improves the overall cloud segmentation performance of an algorithm [44]. In this investigation, the performances of the voting schemes that use all proposed methods and the optimal combination of methods are compared.

2. Datasets and measurements

The features from the IR images are extracted after processing to remove the Sun and atmospheric scattering effects. Due to difficulty accessing the DAQ location, the IR camera's window cannot be routinely cleaned. We implemented a persistent model to remove the effects produced by dust particles and water spots on the window, using standard weather parameters from a nearby weather station [56], prior to the feature extraction. The extracted features from a pixel and its neighboring pixels are cross-validated to find the set of features that increases the segmentation performances in each method.

2.1. Data acquisition system

The proposed segmentation methods utilize a DAQ system equipped with a solar tracker that updates its pan and tilt every second, maintaining the Sun in a central position in the images throughout the day. The IR sensor is a Lepton[1] radiometric camera with an 8 to 14 μm wavelength. The pixels in a frame are emission temperature measurements in centi-kelvin units. Henceforth, the emission temperature is referred to as "temperature" for short. The resolution of an IR image is 80×60 pixels, and the diagonal FOV is $60°$. The DAQ is located on the roof of the UNM-ME building in Albuquerque, NM. The data is publicly accessible in the Dryad repository [60].

The weather parameters that were used to compute cloud height and remove cyclostationary artifacts on the IR images are: atmospheric pressure, air temperature, dew point and humidity. The weather station performs a measurement every 10 min. The data was interpolated to match the IR images samples. The weather station is located at the University of New Mexico Hospital. It is publicly accessible.[2]

2.2. Feature extraction

A pixel of the camera frame is defined by a pair of Euclidean coordinates i,j. The temperature of the clouds in an IR image are $\mathbf{T} = \{T_{i,j} \in \mathbb{R}^+ | \forall i = 1, ..., M, \forall j = 1, ..., N\}$ in Kelvin, and are measured using the radiometric IR camera. The temperature of a particle in the troposphere is roughly a function of the height [61] (other weather variables affect the height of clouds). The height of a pixel in a frame can be approximated using the Moist Adiabatic Lapse Rate (MALR) function [62], that we define as $\varphi : (T^{air}, T^{dew}, P^{atm}) \mapsto \Gamma_{MARL}$ [56]. The weather parameters necessary to compute Γ_{MARL} are: air temperature T^{air}, dew point T^{dew} and atmospheric pressure P^{atm} (measured on ground-level). The height of a cloudy pixel is computed using this formula $H_{i,j} = [T_{i,j} - T^{air}]/\Gamma_{MARL}$. The height of the pixels are defined in kilometers as $\mathbf{H} = \{H_{i,j} \in \mathbb{R}^+ | \forall i = 1, ..., M, \forall j = 1, ..., N\}$ and they are computed using the MALR function.

2.3. Image preprocessing

The DAQ system germanium outdoor lens is cleaned when it rains. However, after the water droplets have evaporated, a water stain is left on the lens. Due to the inconvenience of cleaning the lens in person, we propose implementing a model of the stains caused by dried water droplets. The algorithm to model the outdoor lens begins with a classification model determining the sky conditions in an IR image.

A linear SVC model was trained to distinguish between four classes of sky-conditions: clear-sky, cumulus, stratus or nimbus cloud. The feature vectors of the model include the mean, variance, kurtosis and skewness of the temperatures $T_{i,j}$ in the images, and the clear-sky index values. Consecutive sequences of images from non consecutive days with different conditions were manually labelled for training and validation. The total number of images was 8200. A 5-fold cross-validation of the linear SVC complexity parameter was implemented keeping a balance between classes during each training and validation iteration.

When sky conditions are detected as clear-sky, the IR image is added to the clear-sky set. At the same time, the algorithm forgets the oldest clear-sky image in the set. The clear-sky set contains the last $L = 250$ of clear-sky images. The scatter irradiance produced by dust and water stains on the outdoor germanium lens is the median image computed using all the IR images in the clear-sky set $\mathbf{W} = \{w_{i,j} \in \mathbb{R} | \forall i = 1, ..., M, \forall j = 1, ..., N\}$. The outdoor lens algorithm and the sky conditions classification model are fully detailed in Ref. [56]. In this investigation, we use the obtained temperature of each pixel, after removing the dust and stains. The temperatures are defined as $\mathbf{T}' = \{T'_{i,j} \in \mathbb{R}^+ | \forall i = 1, ..., M, \forall j = 1, ..., N\}$, and the heights are $\mathbf{H}' = \{H_{i,j}' \in \mathbb{R}^+ | \forall i = 1, ..., M, \forall j = 1, ..., N\}$.

The image preprocessing method implemented in this article is introduced in Ref. [56]. The raw temperature of a pixel i,j is processed using a model that combines the effects of the scatter irradiance $\mathscr{S}(\cdot)$ and the direct irradiance from the Sun $\mathscr{D}(\cdot)$. The function of the background atmospheric irradiance $\mathscr{A}(\cdot)$ is,

$$\mathscr{A}(i,j;\mathbf{x}_0,\boldsymbol{\Theta}) = \mathscr{S}(j;y_0,\theta_1,\theta_2) + \mathscr{D}(i,j;\mathbf{x}_0,\theta_3,\theta_4)$$

$$= \theta_1 \exp\left\{\frac{j - y_0}{\theta_2}\right\} + \theta_3 \left\{\frac{\theta_4^2}{\left[(i - x_0)^2 + (j - y_0)^2 + \theta_4^2\right]^{\frac{3}{2}}}\right\}$$

(1)

where the parameters are $\boldsymbol{\Theta} = \{\theta_1, ..., \theta_4\}$ and $\mathbf{x}_0 = \{x_0, y_0\}$ is the position of the Sun in the image. This function is used to model the deterministic component of the irradiance in the IR images.

The optimal parameters of the scatter irradiance model are different in each image. We propose to model these parameters as the predictors of a function whose covariates are: air temperature, dew point, and the Sun's elevation and azimuth angle. To approximate the parameters of the modeling function, the atmospheric background model is optimized in a set of frames with clear-sky conditions from different days of year. The parameters of the Sun's direct radiation model are constants.

After removing both the window model and the atmospheric model from the images, differences of temperature with respect to the tropopause's temperature are defined as $\Delta\mathbf{T} = \{\Delta T_{i,j} \in \mathbb{R} | \forall i = 1, ..., M, \forall j = 1, ..., N\}$. The differences of height are also computed and multiplied by the tropopause's average temperature in the image, estimated using the atmospheric background model. The resulting heights are $\mathbf{H}'' = \{H_{i,j}'' \in \mathbb{R}^+ | \forall i = 1, ..., M, \forall j = 1, ..., N\}$. .

The temperature differences are normalized to 8 bits, $\mathbf{I} = \{i_{i,j} \in \mathbb{N}^{2^8} | \forall i = 1, ..., M, \forall j = 1, ..., N\}$. He aim of the normalization is to extract a feature that simplifies the classification of a model. Through the normalization, information about the feasible minimum temperature of a cloud is added to each pixel. The normalization formula is $i_{i,j} = [\Delta T_{i,j} - \mathbf{min}(\Delta\mathbf{T})]/[(11.5 - 1.52) \cdot 9.8]$, the lowest value is set to 0, and then divided by the clouds' maximum feasible temperature [56]. The feasible temperature is calculated assuming a linear temperature decrease of 9.8K/km in the tropopause [63], and that the average tropopause height is 11.5km at $36°$ latitude north [64]. The average height above sea level is 1.52km in

---

[1] https://www.flir.com/.
[2] https://www.wunderground.com/dashboard/pws/KNMALBUQ473.

Table 1
Type of clouds, percentage of cloud covered and the Sun's position in the horizon in each IR image of the training and test sets.

	Type of Cloud	Cloud Covered [%]	Elevation [°]	Azimuth [°]
Train No. 1	Stratocumulus and Cumulus	38.67	29.69	160.91
Train No. 2	Stratocumulus	28.13	28.70	157.49
Train No. 3	Cirrocumulus and Stratocumulus	36.6	24.43	146.99
Train No. 4	Altocumulus	4.5	31.40	183.09
Train No. 5	Cumulus	37.94	73.52	172.20
Train No. 6	Nimbus	100	29.92	164.81
Train No. 7	Clear-Sky	0	28.94	158.59
Test No. 1	Contrail	38.67	48.76	183.94
Test No. 2	Cumulus	28.13	37.53	149.34
Test No. 3	Altocumulus	12.29	32.10	204.17
Test No. 4	Clear-Sky	0	76.58	190.5
Test No. 5	Altostratus	100	60.07	165.62

Albuquerque, NM.

The velocity vectors were computed applying the Weighted Lucas-Kanade method (WLK) [65,66]. For each two consecutive images $\mathbf{I}^{k-1}$, $\mathbf{I}^k$ of the data set, the velocity vectors are defined as $\mathbf{V}^k = \{\mathbf{v}_{i,j} = (u,v)^k_{i,j} \in \mathbb{R}^2 \mid \forall i = 1, ..., M, \forall j = 1, ..., N\}$. The upper index k denoting the frame is omitted in the rest of the document.

The pixels in the images that form the dataset were manually labelled as clear $y_{i,j} = 0$ or cloudy $y_{i,j} = 1$. The temperature in the background of the images varies. The background temperature is the temperature of the tropopause. For each image, this temperature was first identified, and then used to distinguish pixels that have the image background temperature.

2.4. Feature vectors

To find the optimal feature combination, we propose the validation of different physical features extracted from a pixel, and three sets of neighboring pixels, included as dependent variables in the model.

The first feature vector, $\mathbf{x}^1_{i,j} = \{T_{i,j}, H_{i,j}\}$, contains the raw radiometric temperature of the pixels and the heights computed using the raw temperatures. The second feature vector, $\mathbf{x}^2_{i,j} = \{T_{i,j}', H_{i,j}'\}$, contains the temperature and height of the pixels after removing the artifacts on the IR camera's window. The third feature vector, $\mathbf{x}^3_{i,j} = \{\Delta T_{i,j}, H''_{i,j}\}$, contains the incremental temperatures and heights after removing the Sun's direct radiation and the atmospheric scatter radiation. The fourth feature vector includes the magnitude of the velocity vectors, the normalized increments of temperature, and the increments of temperature; and is defined as $\mathbf{x}^4_{i,j} = \{\text{mag}(\mathbf{v}_{i,j}), i_{i,j}, \Delta T_{i,j}\}$.

To segment a pixel, its feature vectors and those of its neighboring pixels are introduced into the classifier. In the experiments, we define 1st *order neighborhood* feature vector as the set of four

pixels closest to the test pixel i,j, 2nd *order neighborhood* is defined as the eight closest pixels, and term *single pixel* is used when no neighbors are included, that is:

- Single pixel: $\{\mathbf{x}_{i,j}\}$, $\forall i,j = i_1, j_1, ..., i_M, j_N$
- 1st order neighborhood: $\{\mathbf{x}_{i-1,j}, \mathbf{x}_{i,j-1}, \mathbf{x}_{i,j+1}, \mathbf{x}_{i+1,j}\}$.
- 2nd order neighborhood: $\{\mathbf{x}_{i-1,j}, \mathbf{x}_{i,j-1}, \mathbf{x}_{i,j+1}, \mathbf{x}_{i+1,j}, ...$
  $..., \mathbf{x}_{i-1,j-1}, \mathbf{x}_{i-1,j+1}, \mathbf{x}_{i+1,j+1}, \mathbf{x}_{i+1,j-1}\}$.

3. Methods

The methods summarized below can be classified as generative when they have the capacity of generating new samples from a likelihood model, that is, when the model implements a density approximation of the form $p(\mathbf{x}|\mathscr{C}_k)$ where $\mathscr{C}_k$ is the segmentation label of the pixel. Discriminative models do not have the ability to generate data since they implement a direct approximation of the posterior $p(\mathscr{C}_k|\mathbf{x})$.

3.1. Generative models

Generative models are either Maximum Likelihood (ML), or Maximum A Posteriori (MAP) methods. When generative models use an input feature structure, together with the use of an energy function for the probabilistic modeling of data (Ising model), they are generally known as MRF models. We summarize below the discriminant analysis, which applies ML inference, GMM and k-means clustering, and supervised and unsupervised MRF methods, with MAP inference.

3.1.1. Discriminant analysis

GDA and NBC are both supervised learning methods, because the training dataset input features $\mathbf{x}_i$ are paired with a label $\mathscr{C}_k$. As we assume that the prior in these models is uniform, the inference applied is ML.
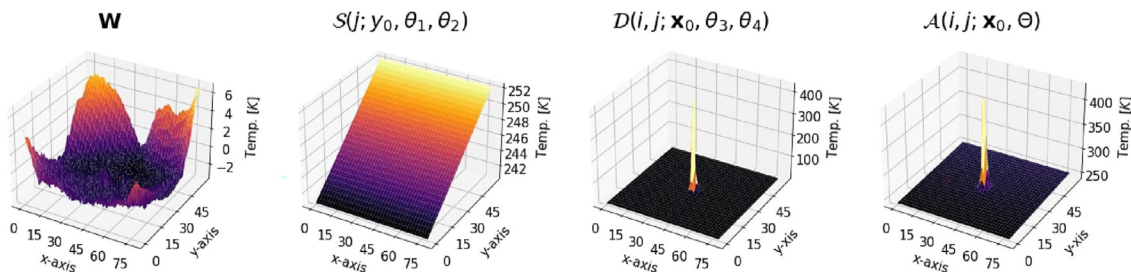

Fig. 1. Models applied to preprocess the radiometric IR images. The models correspond to the IR image in Fig. 2. The first image is the atmospheric scattered irradiance. The second is the Sun's direct irradiance. The third is the atmospheric background model that combines both previous models. The last image is the outdoor lens scattering effect model.
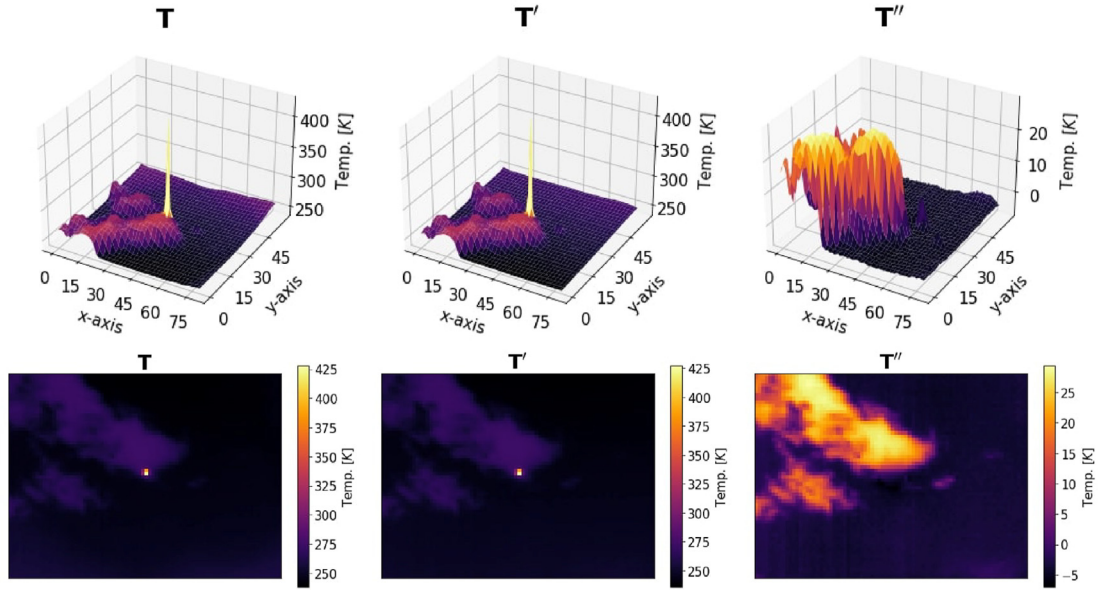
Fig. 2. The IR images are shown in a 3-dimensional surface graph in the top row and in the figures in the bottom row. The figures in the first column are the raw pixels obtained from the radiometric IR camera. The figures in the second column show the processed IR images, removing the atmospheric background model. The figures in the third column are the IR images after processing to remove the atmospheric background and the outdoor lens scattering effect. The applied models are shown in Fig. 1.

Gaussian Discriminant Analysis. GDA obtains the posterior probability of $y_i = \mathscr{C}_k$ given a set of features $\mathbf{x}_i \in \mathbb{R}^d$ when applying the Bayes theorem [67], where a prior is chosen over the classes, and a Gaussian likelihood is used for the observations. The posterior of class $\mathscr{C}_k$, where $k \in \{1, ..., K\}$ are possible classes, is maximized by the Bayes' rule with the expression $p(\mathbf{x}_i) \propto p(\mathscr{C}_k)p(\mathbf{x}_i|\mathscr{C}_k)$.

The corresponding means $\boldsymbol{\mu}_k \in \mathbb{R}^d$ and covariance matrices $\boldsymbol{\Sigma}_k \in \mathbb{R}^{d \times d}$ are estimated with the samples that have assigned class $\mathscr{C}_k$ and d is the sample dimension, i.e, the number of features in vector $\mathbf{x}_i$.

Naive Bayes Classifier. The NBC applies the Bayes theorem, similarly to a ML classifier, but it computes a likelihood by assuming that all features are independent. Therefore, the corresponding Gaussian likelihood is approximated by a product of a univariate Gaussian distribution per each dimension of the observation $\mathbf{x}$ [68].

3.1.2. Clustering

The GMM and k-means are unsupervised learning algorithms. Their respective objective functions group the samples in clusters represented by conditional likelihood functions, and then a posterior distribution for each class $\mathscr{C}_k$ is computed with the likelihood and a prior distribution of the labels. Thereby, the inference level applied is MAP. K-means can be considered as a simplification of the GMM.

Gaussian Mixture Model. The GMM assumes a known number K of possible latent variable values. For each one, a Gaussian distribution is constructed. Initial values are proposed for these parameters and for the priors of the classes. The Expectation Maximization (EM) algorithm [68] is used to iteratively adjust all these parameters. In the E step, posteriors $p(\mathscr{C}_k|\mathbf{x}_i)$ are computed for all samples. In the M step, mean μ_k is computed by averaging all samples weighted by their corresponding posterior. The covariance is computed similarly. The priors are computed by averaging the posteriors. Once these statistics are computed, the E step is repeated, until convergence.

k-means. K-means is a simplification of a GMM, where co-variances $\boldsymbol{\Sigma}_k = \mathbf{I}_{d \times d}$, are assumed to be constant [68,69]. The posterior distribution for the latent variables is 1 for the class whose

mean is closer to the sample and zero otherwise.

3.1.3. Markov Random Fields

In a MRF, pixels are grouped in cliques of a graph $\mathscr{G}$ where a clique is a set of nodes which are neighbors of each other given a definition of neighborhood. For the problem at hand, a node represents a pixel, and a neighborhood is defined in terms of pixel proximity. The prior probability of a pixel's class is a normalized exponential of an energy function. This represents the *energy* of a clique.

The energy function of a MRF is composed of ϕ, or the joint distribution of a class, and ψ, the potential energy of the system's configuration [70].

$$\mathscr{E}(y_i, \mathbf{x}_i) = \sum_i \phi(\mathbf{x}_i, y_i) + \sum_{ij} \psi(y_i, y_j), \quad (2)$$

In graph G, a sample i has a set of neighboring pixels, and each neighboring sample j has class y_j. A sample $\mathbf{x}_i$ is classified using the Bayes' theorem,

$$p(y_i = \mathscr{C}_k|\mathbf{x}_i, \boldsymbol{\theta}_k) \propto p(\mathbf{x}_i|y_i = \mathscr{C}_k, \boldsymbol{\theta}_k)p(y_i = \mathscr{C}_k). \quad (3)$$

where $p(\mathbf{x}_i|y_i = \mathscr{C}_k, \boldsymbol{\theta}_k) = \mathscr{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ with $\boldsymbol{\theta}_k = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$. The log-likelihood of class $\mathscr{C}_k$ is defined as $\phi(\mathbf{x}_i, y_i)$ in the energy function (2). The prior is

$$p(y_i) = \frac{1}{Z}\exp(-\psi(y_i)) = \frac{1}{Z}\exp\left(\sum_{i,j \in \Omega_\ell} y_i \beta y_j\right), \quad (4)$$

where the potential function has been factorized in cliques of a graph $\mathscr{G}$ by applying the Hammersley−Clifford theorem [71]. A clique is defined as a set of nodes that are all neighbors of each other [68]. With the above model, a posterior can be constructed to classify the pixels. If there are no labelled images, then unsupervised inference of the class parameters in a MRF model can be performed using the ICM algorithm [48]. The standard optimization goes through all the pixels calculating their potential and
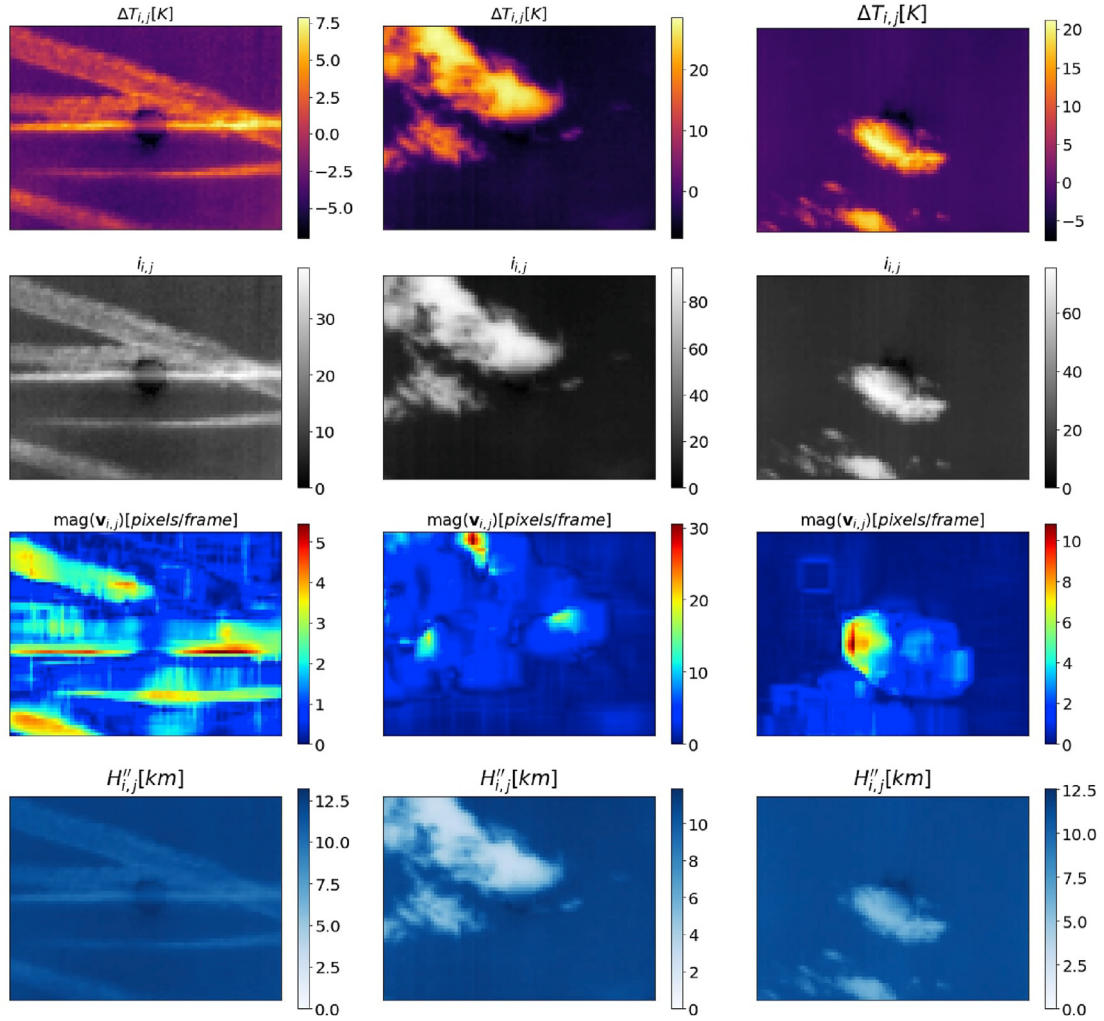
Fig. 3. This figure shows the features extracted from three test images. The test images are organized in columns. The images in the first row show the normalized intensity of the pixels. The images in the second row show the magnitude of the velocity vectors. The images in the third row show the increments of temperature with respect to the height of the tropopause. The images in the fourth row show the height of the clouds. The last row shows the test images in which the clouds were manually segmented.

classifying them in each iteration of the algorithm. The computational cost of this method is high, but we can assume that it is not necessary to evaluate the pixels whose state has high energy, because their classification will not change. The computational cost can be reduced by sampling the pixels that are likely to be misclassified, and applying the optimization procedure only to them.

We propose to optimize the configuration of the pixels in an IR image applying the SA algorithm [72] to the MRF models [73]. The SA algorithm is applied on the implementation, after the inference of the class distributions. Temperature parameter T in the SA algorithm is linearly cooled down $T^{t+1} = \alpha T^t$ at each iteration t with an acceptance rate α. The optimal parameter α is a trade off between accuracy and speed.

3.2. Discriminative models

The discriminative algorithms in this work are based on kernel models [57], where the data is implicitly transformed into a Hilbert space $\mathcal{H}$ of higher dimensionality (possibly infinite) with a dot product expressible as a positive definite function of the input data [74]. Kernel Learning uses the generalized Representer Theorem [75], which states that, under mild conditions, any machine can be represented by a linear combination of dot products between the

training and test data (which is called a *dual* formulation). This implies the use of a matrix containing the N^2 kernel dot products between data, which can be overwhelming due to the high quantity of data to be used during the training in the present problems. Therefore, we use explicit basis functions for the nonlinear transformation into $\mathcal{H}$. The matrices to be manipulated have the same dimension of this space.

The transformation is a polynomial expansion of the elements of $\mathbf{x}$. The expansion is defined as $\phi : \mathcal{X} \mapsto \mathcal{P}^n$, where n is the order of the expansion and $\mathcal{P}^n = (n + d)!/n!$. The polynomial expansion of the dataset $\mathcal{D} = \{\Phi, \mathbf{y}\}$, is defined in matrix form as

$$\Phi = [\, \phi(\mathbf{x}_1) \quad \cdots \quad \phi(\mathbf{x}_N) \,] \in \mathbb{R}^{\mathcal{P}^n \times N}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}, \tag{5}$$

where $y_i \in \{0, 1\}$ which are labels for a clear or cloudy pixel, respectively. The polynomial expansion is used in the primal formulated kernel for RR, SVC and GPC is defined such as
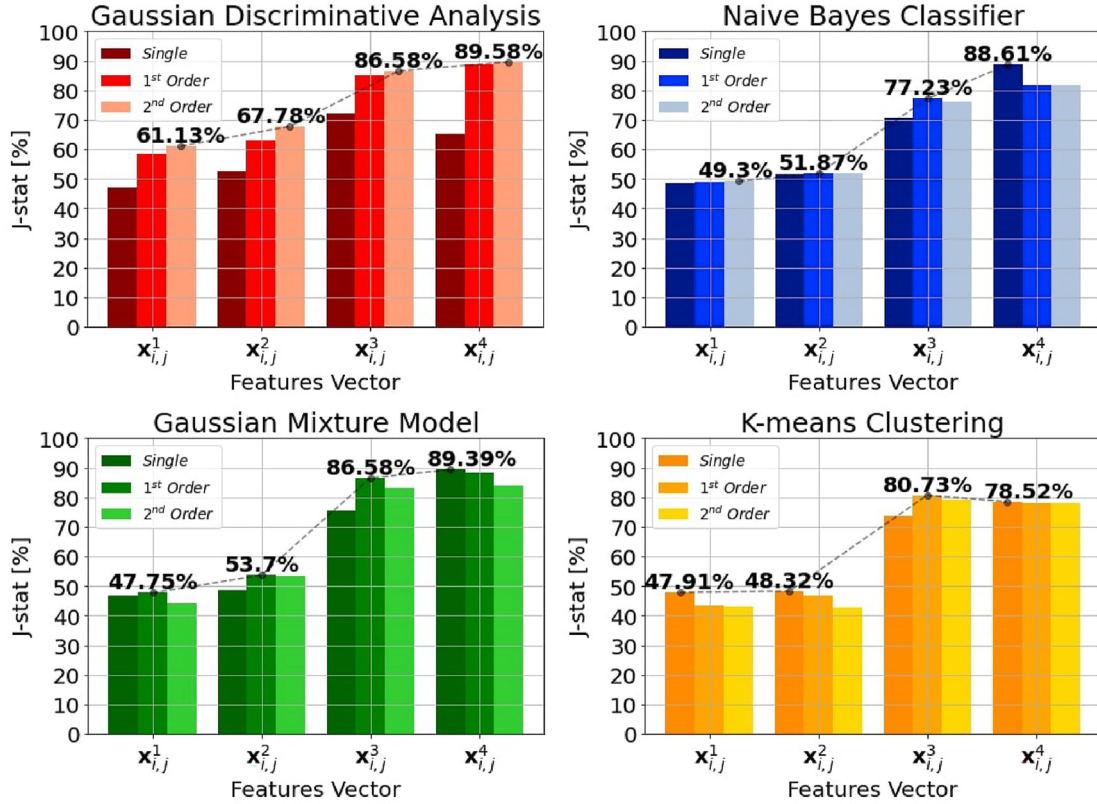
Fig. 4. The graph shows the j-statistic achieved by the generative models. The color of the bars in the graph indicate the order of neighborhood from dark to light. The neighborhoods are organized from the left to right within the groups of bars. This corresponds with the order of the feature vectors used in the model.

$$\phi(\mathbf{x}_i) = \begin{bmatrix} 1 \cdots a_j x_j \cdots a_{j,k} x_j x_k \cdots a_{j,k,l} x_j x_k x_l \cdots \end{bmatrix}^\top \in \mathbb{R}^{\mathscr{P}^n}, \quad (6)$$
$$\forall j, k, l \cdots = 1, \ldots, D$$

where scalars $a_j, a_{j,k}, a_{j,k,l}, \cdots \in \mathbb{R}$ are chosen so that the corresponding dot product in the space can be written as

$$\phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_i) = \begin{bmatrix} 1 + \mathbf{x}_i^\top \mathbf{x}_i \end{bmatrix}^n. \quad (7)$$

which is the well known polynomial kernel of order n.

3.2.1. Ridge Regression

The RRC is a minimum mean squared error method with quadratic norm regularization applied on the parameters $\mathbf{w}$,

$$\min_{\mathbf{w}} \sum_{i=1}^{N} (\mathbf{y} - \mathbf{w}^\top \mathbf{\Phi})^2 + \gamma ||\mathbf{w}||_2. \quad (8)$$

where γ is the regularization parameter, and it requires cross validation. The solution is simply $\overline{\mathbf{w}} = (\mathbf{\Phi}\mathbf{\Phi}^\top + \gamma \mathbf{I})^{-1} \mathbf{\Phi}^\top \mathbf{y}$. The output is passed through a sigmoid function in order to provide it with probability mass properties, i.e

$$p(y=1|\phi(\mathbf{x}), \mathscr{D}) = \frac{1}{1 + \exp(-\overline{\mathbf{w}}^\top \phi(\mathbf{x}))} \quad (9)$$

3.2.2. Primal solution for support vector machines

Several approaches to solve the SVC in the primal space have been proposed, as the iterative re-weighted least squares method

[76], or by directly solving the following quadratic problem [77,78],

$$\min_{\mathbf{w}} \frac{1}{2} ||\mathbf{w}||_2 + C \sum_{i=1}^{N} (\max[0, 1 - y_i \mathbf{w}^\top \phi(\mathbf{x}_i)])^2, \quad (10)$$

where $C > 0$ is a penalty term, $x_i(\mathbf{w}, \mathbf{x}_i, y_i)$ is the loss function, and in our model the norm is a L_2. This is a maximum margin problem [77]. The complexity parameter C has to be cross-validated.

The original formulation of the linear SVC does have not a probabilistic output. However, the distance from a sample to the hyper-plane can be transformed to a probability measure,

$$p(\mathscr{C}_1|\phi(\mathbf{x}_*), \mathscr{D}) = \frac{1}{1 + \exp(-\overline{\mathbf{w}}^\top \phi(\mathbf{x}_*))} \quad (11)$$
$$p(\mathscr{C}_2|\phi(\mathbf{x}_*), \mathscr{D}) = 1 - p(\mathscr{C}_1|\phi(\mathbf{x}_*), \mathscr{D}),$$

using the sigmoid function, similarly to the proposed RRC for classification.

3.2.3. Primal solution for Gaussian Processes

When formulated in the primal, a GPC is known as Bayesian logistic regression [59,68,79]. A posterior over the primal parameters $\mathbf{w}$ is computed as

$$p(\mathbf{w}|\mathscr{D}) \propto p(\mathbf{y}|\mathbf{\Phi}, \mathbf{w})p(\mathbf{w}). \quad (12)$$

The likelihood function of the model is $p(y_i|\mathbf{\Phi}, \mathbf{w}) = \prod_{i=1}^{N} \hat{y}_i^{y_i}(1 - \hat{y})^{1-y_i}$, where $\hat{\mathbf{y}} = [\hat{y}_1 \ldots \hat{y}_N]^\top$ are the predictions. The prior is Gaussian $p(\mathbf{w}) \sim \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$, but the posterior is not Gaussian. In contrast to Bayesian linear regression, this approach does not have an analytical solution. The Laplace approximation is
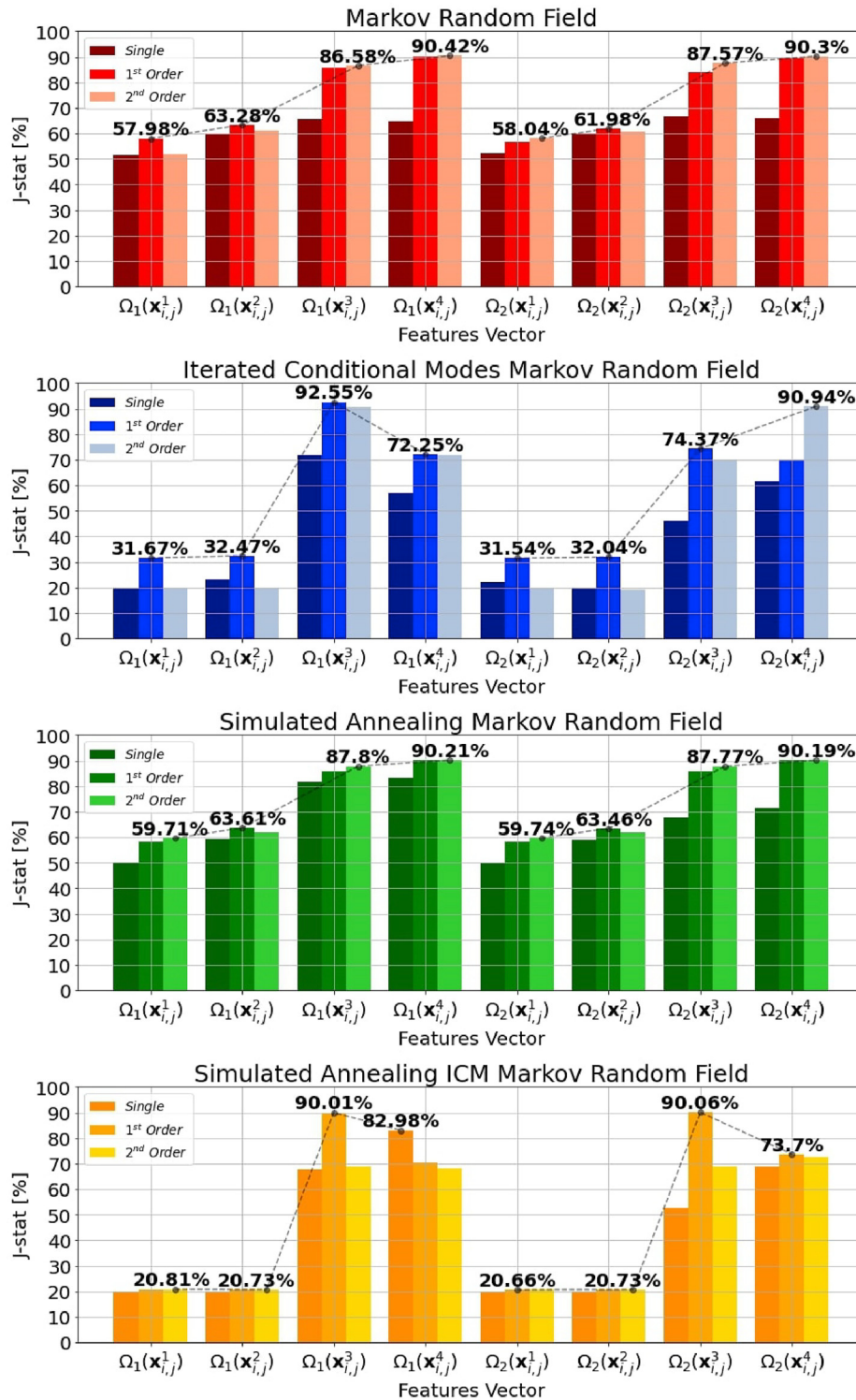
Fig. 5. The graphs show the j-statistic achieved by the MRFs using different cliques in their potential function. The four feature vectors are organized in groups of three bars. There are two groups of feature vectors: those with a potential function of 1st order cliques $\Omega_1(\cdot)$, and those with a potential function of 2nd order cliques $\Omega_2(\cdot)$.

applied to assume that the posterior is Gaussian $q(\overline{\mathbf{w}}) \sim \mathcal{N}(\mathbf{w}|\overline{\mathbf{w}}, \Sigma_{\mathbf{n}})$. The optimal set of parameters are found maximizing the marginal log-likelihood via numerical gradient.

4. J-statistic

The Younde's j-statistic or Younde's Index is a test to evaluate the performances of a binary classification [80], that is defined as,

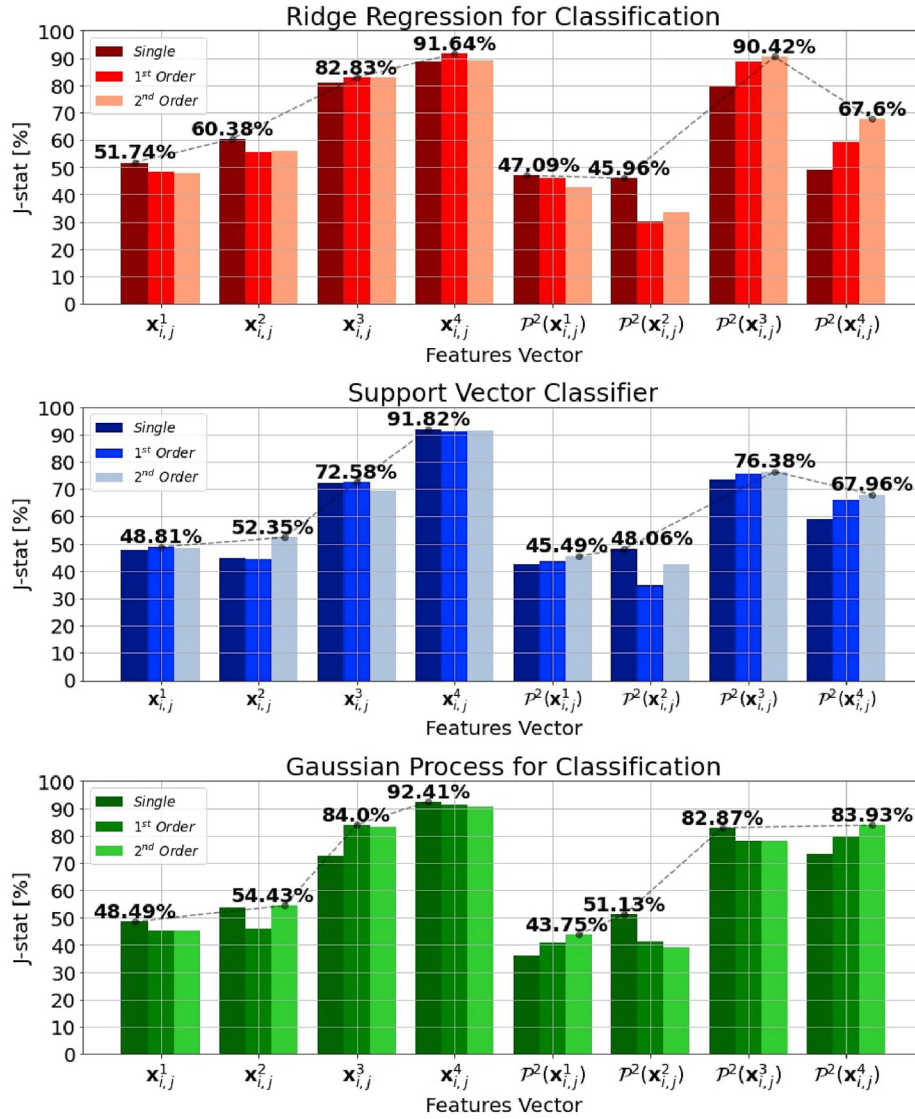$$J = sensitivity + specificity - 1. \tag{13}$$

Fig. 6. The graphs show the j-statistics achieved by the discriminative models. The feature vectors are organized in groups. The bars in the same group from dark to light are: features extracted from a single pixel, a 1st order neighborhood and a 2nd order neighborhood. When a polynomial expansion of the second order is applied to the feature vectors, it is denoted as $\mathscr{P}^2(\cdot)$.
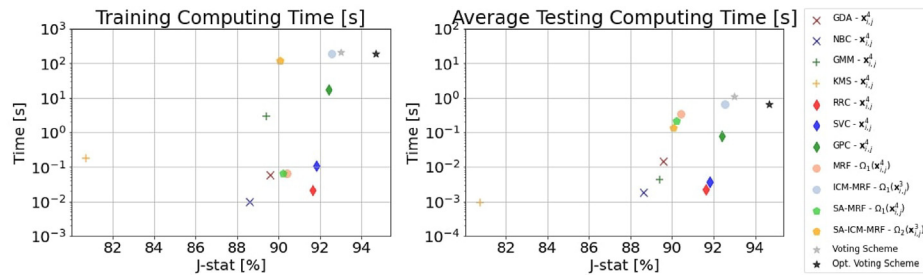

Fig. 7. Left: Computing time of each model during training. Right: Average computing time during the segmentation in the test subset. The legend displays the optimal feature vectors, neighborhood order, polynomial expansion and cliques of each model.

The entries on the confusion matrix are used to compute the sensitivity,

$$sensitivity = \frac{TP}{TP + FN},$$

(14)

where *TP* and *FN* are the true positives and false negatives, and the
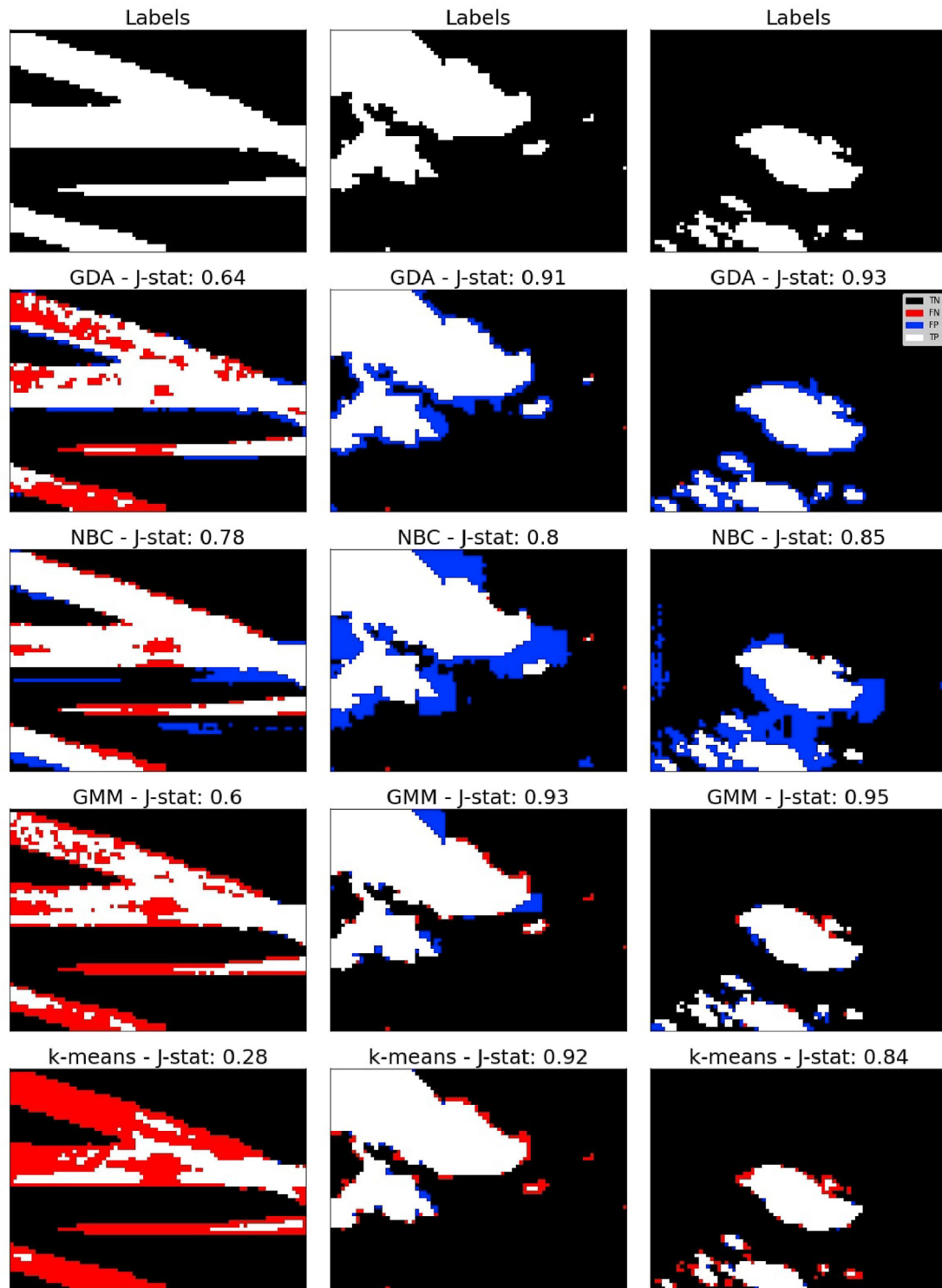
Fig. 8. Three images from the test organized in columns. The images in each row show the segmentation performed by a generative model. The higher j-statistic was achieved by the NBC in the first image, and the GMM in the second and third images.

specificity is,

$$specificity = \frac{TN}{TN + FP},$$ (15)

where *TN* and *FP* are the true negatives and false positives. It is different than the accuracy score of a binary classification, which is

also obtained using the entries of the confusion matrix, and that it is,

$$ACC = \frac{TP + TN}{TP + FP + TN + FN}.$$ (16)

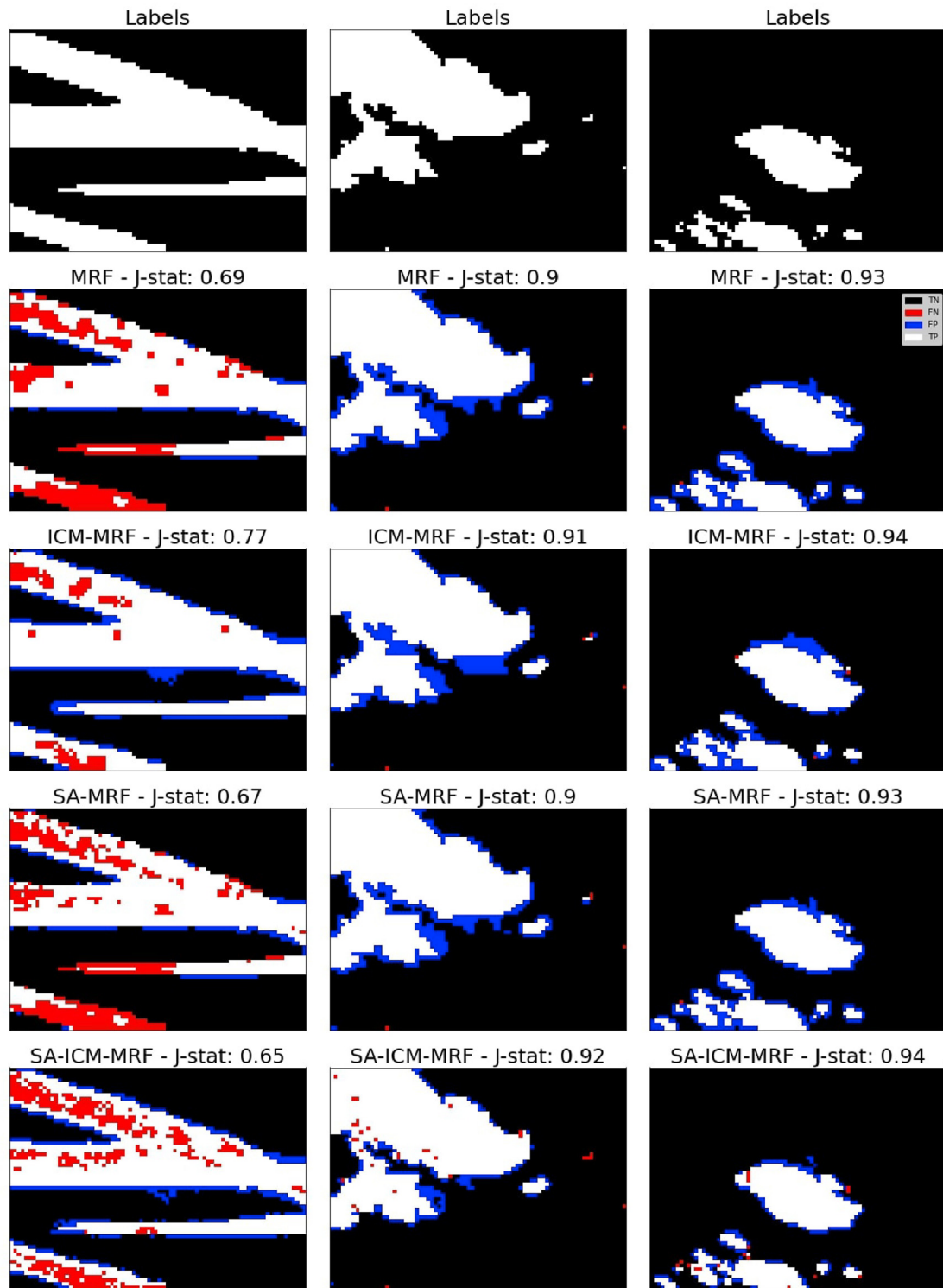As the optimized loss function is different in each model, we

Fig. 9. Three images from the test subset organized in columns. The images in each row show the segmentation performed by a MRF model. The highest j-statistic was achieved by ICM-MRF in the first image, SA-ICM-MRF in the second, and ICM-MRF and SA-ICM-MRF in the third image.

propose to define a prior λ, which has to be cross-validated for each one of models, and has an optimal value for each classification function,

$$
\begin{aligned}
p(\mathscr{D}|\mathscr{C}_k) &= \frac{p(\mathscr{C}_k|\mathscr{D})p(\mathscr{C}_k)}{p(\mathscr{D})} \\
&\propto p(\mathscr{C}_k|\mathscr{D})p(\mathscr{C}_k) \\
&\propto p(\mathscr{C}_k|\mathscr{D})\lambda
\end{aligned}
\tag{17}
$$

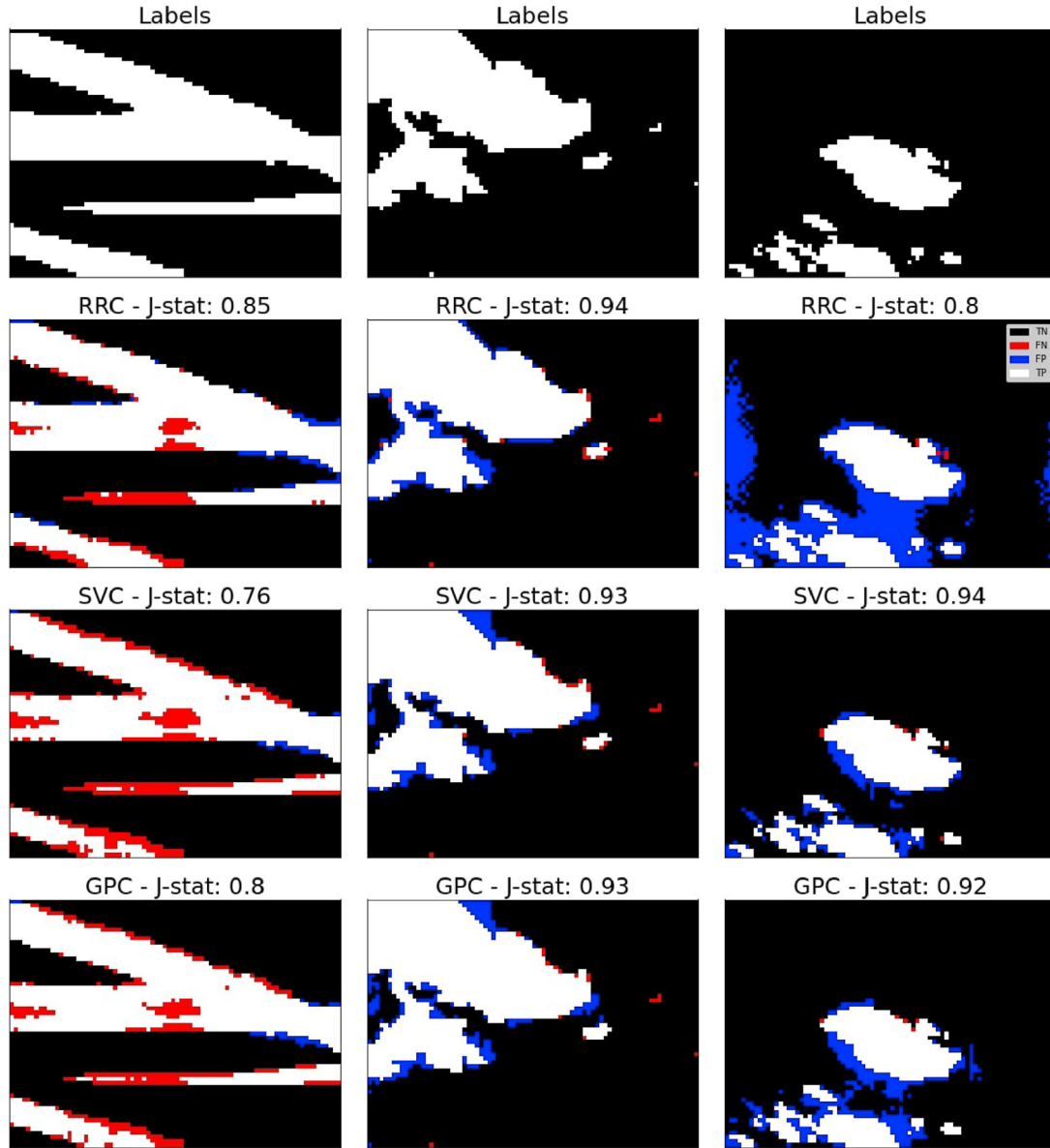so the maximized loss function is the same in all the models. The

Fig. 10. Three images from the test subset are organized in columns. The images in each row show the segmentation performed by the discriminative models. When segmenting the images, a higher j-statistic was achieved by RRC (in the first and second image), and SVC (in the third image).

classification probabilities are defined as $p(\mathscr{D}|\mathscr{C}_1) = p(\mathscr{C}_1|\mathscr{D})\lambda$, and $p(\mathscr{D}|\mathscr{C}_2) = 1 - p(\mathscr{D}|\mathscr{C}_1)$. The j-statistic score is maximized finding the optimal binary classification λ threshold. For that, the j-statistic is applied to the conventional Receiver Operating Characteristic (ROC) analysis [81], and it is computed at each point of the ROC. We propose to use the maximum value of j-statistic in the ROC curve as the optimal point.

After the cross-validation of the virtual prior λ, a class $\mathscr{C}_k$ is assigned to a sample $\mathbf{x}_*$ following this criteria,

$$\widehat{y}_* = \underset{k}{\operatorname{argmax}} \, p(\mathscr{C}_k|\mathbf{x}_*, \mathscr{D})\lambda, \tag{18}$$

which is a MAP estimation.

5. Experiments

The selected samples constitute the dataset used in the cross-

validation and testing of the segmentation models. The samples include different days in all four seasons. The sample images include partially cloudy, fully clear-sky and fully covered sky conditions. The images were captured at different hours of the day, so the Sun's elevation and azimuth angle are different. Therefore, the atmospheric background model is different in all of the images. The dataset is composed of different types of clouds found at different heights in the tropopause. We found that the most difficult clouds for the models to classify are cirrus stratus, which are also included in the dataset. Artificially created clouds like contrails are also included in testing set. Contrails are highly difficult for the models, as the scattering effect is similar to that produced by cirrus clouds. The dataset is composed of 12 images with labels (see Table 1), amounting to a total of 57,600 pixels. They are organized chronologically and divided into training (earlier dates) and testing set (later dates). The training set has 7 images, which are 33,600 pixels in total. The testing set has the remaining 5 images, which are
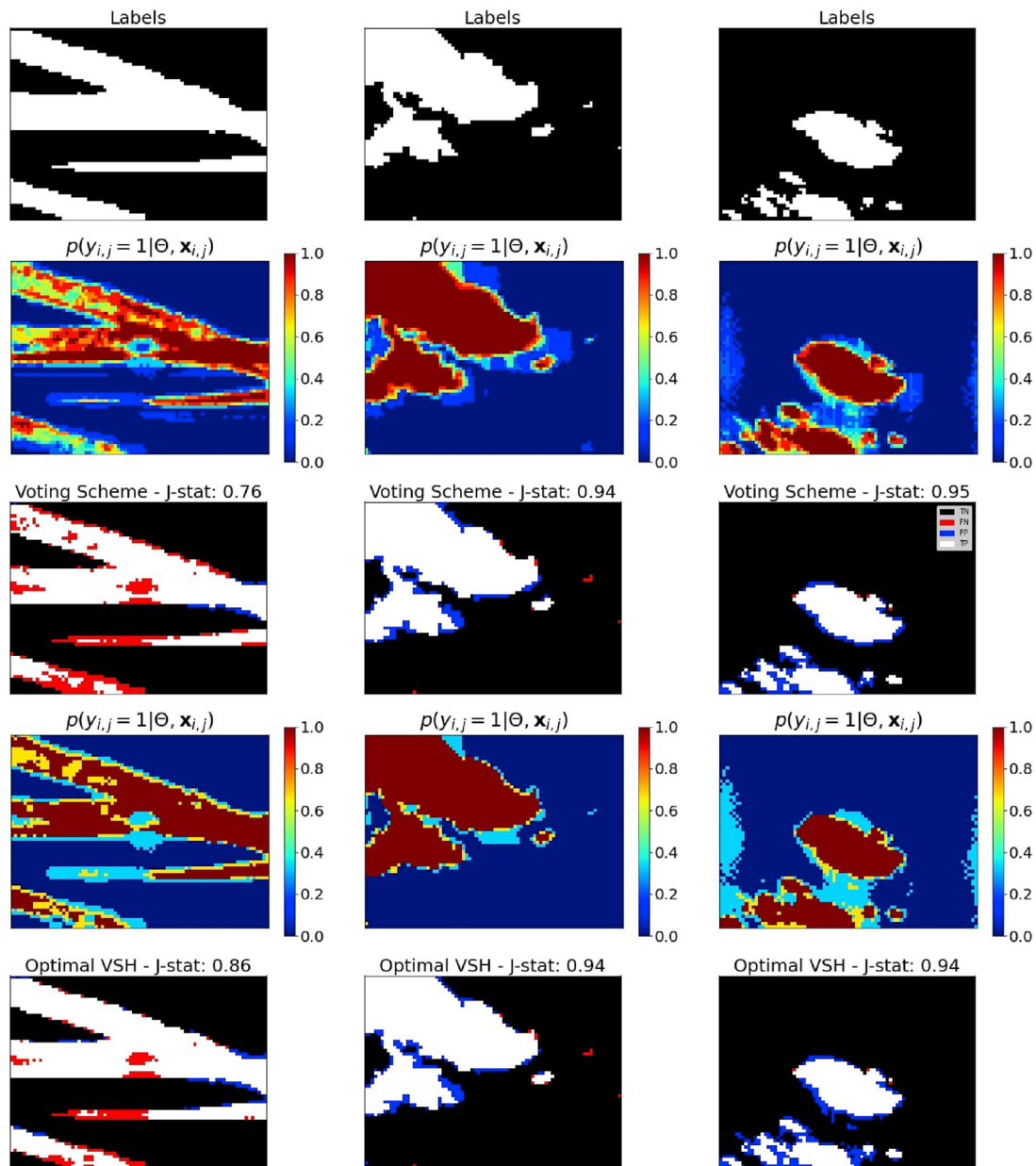
Fig. 11. Three different test images. First and second rows: results of the voting scheme. The first row displays the probability of a pixel belonging to a cloud. The second row shows the segmentation performed by the voting scheme. Third and fourth rows: probability of a pixel belonging to a cloud and the segmentation of the optimal voting scheme (VSH).

24,000 pixels. The training set contains 5 images with clouds, 1 image with clear-sky, and another one with covered sky conditions. The testing set has 3 images with clouds, 1 with clear-sky, and 1 with covered sky conditions.

The Leave-One-Out (LOO) method is implemented in the cross-validation of the parameters. In this method, the training samples are left out for validation one at a time, while the rest of the training samples are used to fit the model. In our problem, the training samples are the images in the training set, so a training image is used for validation while the others are used for training the model.

The cross-validation is done using a high performance computer. Each validation sample (in the LOO routine) runs on a different CPU, and 7 CPUs are necessary for each experiment. When the LOO routine is finished, the results are communicated to the main node, and a new set of hyperparameters and virtual prior λ are validated. This procedure is repeated until all possible

combinations of hyperparameters and virtual priors are validated. The LOO routine runs in multiple experiments at the same time. Each experiment has a combination of feature vectors, neighborhoods, polynomial expansions (in the discriminative models) and cliques (in the MRF models). All CPUs are operating at full capacity and are only inactive during the waiting time (i.e. until all jobs of the LOO routine are finished).

The cross-validation is computationally expensive due to the amount of training samples, but running the LOO routine and the experiments in parallel reduces the training time by several orders of magnitude. The testing times are obtained when running each segmentation model in a single CPU.

Exploratory results showed that the features that work best are those in vectors $\mathbf{x}_{i,j}^3$ and $\mathbf{x}_{i,j}^4$. All possible combinations were tested, but none produced any improvement in the classification

performance, with the exception of those in $\mathbf{x}_{i,j}^4$. However, the original features require preprocessing to achieve reasonable performances. This is shown in the classification results obtained by $\mathbf{x}_{i,j}^1$ and. $\mathbf{x}_{i,j}^2$.

In the generative models, NBC and k-means clustering do not have hyperparameters. The GDA and GMM have the covariance matrix regularization term γ which has to be cross-validated. In the k-means clustering, the feature vectors were standardized $\overline{\mathbf{x}}_{i,j} = [\mathbf{x}_{i,j} - \mathbf{E}(\mathbf{X})]/\mathbf{Var}(\mathbf{X})$. The rest of the models neither required normalization nor standardization of the feature vectors.

In the MRF models, the cliques potential β in Eq. (4) was cross-validated in all the models. The supervised MRF have the covariance matrix regularization term γ which was cross-validated. The unsupervised ICM-MRF is computationally expensive, so the regularization term of the covariance matrix was set to $\gamma = 1$. In the supervised MRF with SA in the implementation, the cross-validated parameters were the regularization term of the covariance matrix γ, and the cooling parameters α. In the unsupervised MRF trained with the ICM algorithm (using the SA algorithm in the implementation), the parameters of the regularization term of the covariance matrix and cooling were set to $\gamma = 1$ and $\alpha = 0.75$.

In the discriminative models, the RRC has the regularization γ in Eq. (8) that has to be cross-validated. The SVC has the complexity term C of the loss function in Eq. (10). The hyperparameters of the GPC are the prior mean $\boldsymbol{\mu}_0$ and the covariance matrix $\boldsymbol{\Sigma}_0$. The prior mean and covariance matrix are simplified to $\boldsymbol{\mu}_0 \triangleq 0$ and $\boldsymbol{\Sigma}_0 \triangleq \gamma\mathbf{I}_{D\times D}$, so only the parameter γ is cross-validated.

In addition to each set of hyperparameters, all models have a virtual prior λ that corrects possible class-imbalances in Eq. (17). The hyperparameters and the virtual prior λ have to be cross-validated. A set of hyperparameters define the ROC curve, and the virtual prior λ is used to find the optimal j-statistic along this curve with the predicted probabilities of each class for each combination. The validation j-statistic is the average of the j-statistics obtained in each LOO cross-validation loop. The model selection criteria is the highest validation j-statistic.

The experiments were carried out in the Wheeler high performance computer of the UNM-CARC, which uses a SGI AltixXE Xeon X5550 at 2.67 GHz with 6 GB of RAM memory per core, 8 cores per node, 304 nodes total, and runs at 25 theoretical peak FLOPS. Linux CentOS 7 is installed.

6. Discussion

The segmentation performed on three testing images by the generative models are shown in Figs. 7–9. The NBC and GDA are both discriminant analysis and supervised learning models (Fig. 8). The k-means and GMM are unsupervised learning methods (Fig. 8). The MRF and SA-MRF are supervised learning models and ICM-MRF and SA-ICM-MRF are unsupervised learning models. The SA algorithm is implemented to speed-up the MRF and ICM-MRF convergence. When MRF models use the SA algorithm, the segmentation is not so uniform (Fig. 9). The cooling mechanism of the SA algorithm ends the optimization before the segmentation has converged to a state of higher energy. The discriminative models used are the RRC, SVC and GPC (Fig. 10). These were solved in the primal formulation so their performances are feasible for real-time cloud segmentation (see Fig. 7). The performances of the models are compared in terms of j-statistic vs. training computing time vs. average computing time in testing. The j-statistic is evaluated with the images in the testing subset. The computing time is measured in seconds. The time in the y-axes of the graphs shown in Fig. 7 are displayed in logarithmic scale. The highest j-statistic is achieved by

the unsupervised MRF, but the training and the average testing computing time are the largest. NBC and RRC have the lowest training times. In the implementation, the k-means, NBC and RRC have the lowest computing time. If we have considered all of this information, the most suitable model would be one of these model.

The unsupervised MRF model (ICM-MRF) achieved the highest j-statistic in testing among generative and discriminative models. The ICM-MRF model uses the feature vector $\mathbf{x}^3$ with a 1st order neighborhood and the set of cliques Ω_1 in the prior. The classification performance of the model decreased when optimized using the SA algorithm, but the average testing time was faster (Fig. 5). The MRF models that use a prior potential function lead to the largest training and average testing computational time. When only generative models without the prior potential function are considered (NBC, GDA, k-means and GMM), the GDA has the highest j-statistic with the feature vector $\mathbf{x}^4$ of a 2nd order neighborhood (Fig. 4). However, if the trade-off between average testing time and j-statistic is considered, the most suitable generative model is the GMM with a feature vector $\mathbf{x}^4$ of a single pixel neighborhood. The generative models which include a simplification of the covariance matrix and that do not use a prior potential function (NBC and k-means) yield the fastest average testing time among all classification models (without considerably decreasing the j-statistic). However, these models have the lowest j-statistic among all the models implemented. Fig. 6 shows the discriminative models' j-statistics. The average testing time is lower than that obtained by the generative MRF models, but the j-statistic is higher than that obtained by the generative models without the potential function. The polynomial expansion yields overfitting in all the discriminative models. The discriminative model that achieved the highest j-statistic is the linear GPC with the feature vector $\mathbf{x}^4$ of a single pixel neighborhood. As seen in Fig. 7, the RRC and SVC are the most suitable methods, as they offer the best compromise between average testing times and accurate segmentation.

The results in Figs. 4–6 show the importance of the feature extraction method in cloud image segmentation (see Fig. 3). The extraction of features makes it easier for the models to differentiate between cloudy and clear-sky pixels, because the distance between feature vectors of different classes increases in the feature space. Through feature extraction, the feature vectors of the same classes group together forming clusters in the feature space. Without extracting features correctly, the feature vectors from both classes (cloudy and clear-sky) are grouped in a single cluster, making it difficult to perform a classification. When the magnitude of velocity vectors are included in the feature vectors, combined with temperature increments and normalized temperature increments, the segmentation models achieved a higher j-statistic. The addition of features from neighboring pixels to the feature vectors improves the performance in some of the models.

When the raw temperature and height are used, all models have poor performance. However, when the images are preprocessed with the outdoor germanium camera window model and the atmospheric model (see Fig. 2), the ICM-MRF reaches a reasonable performance of 92.55% at the expense of a high computational cost of 641 ms per image in testing. The performance of discriminative methods with this set of features is lower, ranging between 72.58% and 84%. When velocity vectors are added to the features, the discriminative methods achieve a similar performance as the ICM-MRF, with computational times of 2.2 ms (RRC), 3.7 ms (SVC) and 77 ms (GPC). The best compromise is the SVC, which is 150 times faster than the ICM-MRF with a small difference in accuracy. The image preprocessing and feature extraction time is 0.1 ms for $\mathbf{x}^1$, 4.7 ms for $\mathbf{x}^2$, 99.9 ms for $\mathbf{x}^3$ and 1079 ms for $\mathbf{x}^4$. When preprocessing time is added to the segmentation time, the average time required by the ICM-MRF is 740.9 ms. This is faster than the

average time required by the discriminative models: 1081 ms (RRC), 1083 ms (SVC) and 1156 ms (GPC).

A voting scheme using the predictions from the models displayed in Figs 8, 9 and 10 (not including SA-MRF and SA-ICM-MRF) achieved higher j-statistic but have a higher computing time. The j-statistic is 93%, see Fig. 11. The combination of the RRC, SVC and ICM-MRF lead to the best j-statistic. The optimal voting scheme reached a j-statistic of 94.68% in testing (see Fig. 11). The voting scheme's training and testing times are the sum of each method's respective computing times. When the models are trained and tested in parallel, the voting scheme's training and testing times are that of the slower models.

7. Conclusion

This investigation seeks to find the optimal methods for real-time ground-based IR cloud segmentation through image pre-processing and feature extraction. Preprocessing was applied to remove underlying cycle-stationary processes, and feature extraction was used to compute cloud height and velocity. The results show that cloud segmentation in ground-based IR images is not only feasible, but achieves high performance in real-time applications. Ground-based IR cameras perform better than visible ones in poor light conditions. We implement a preprocessing algorithm that uses physical features extracted from IR images. The j-statistic is proposed to independently measure the accuracy of the classification in each classes.

Preprocessing the ground-based IR images using the window and atmospheric models leads to an overall performance improvement. Simplification of the covariance matrix reduces the computing time, but the j-statistic achieved is lower than that of the models using the full covariance matrix. Adding the features of neighboring pixels to the feature vectors yields an increase in segmentation performance in some cases. The discriminative models formulated in the primal result in feasible segmentation models for real-time application. MRF models remove possible outliers using cliques from neighboring pixels. This increases the overall performance of the generative models when trained with unsupervised and supervised algorithms. The optimal voting scheme achieved the best j-statistic. However, the implementation computing time might be slow for real-time applications when not run in parallel.

Further investigations may focus on segmentation in multiple layers of clouds. The clouds in each layer may be segmented into different classes. An algorithm can be trained to detect multiple layers of clouds when clouds have different heights or directions. In this way, the extraction of features may be performed independently in each one of the cloud layers. A multiple cloud layer segmentation algorithm will reduce the noise when extracting features. This algorithm may be implemented to increase the performance of ground-based intra-hour GSI forecasting.

CRediT authorship contribution statement

Guillermo Terrén-Serrano: Software, Formal analysis, Methodology, Investigation, Data curation, Writing — original draft, Visualization. **Manel Martínez-Ramón:** Writing — review & editing, Conceptualization, Supervision, Project administration, Resources, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] W. Zappa, M. Junginger, M. van den Broek, Is a 100% renewable european power system feasible by 2050? Appl. Energy 233—234 (2019) 1027—1050.
[2] U. S. E. A. I, Annual Energy Outlook 2021, 2021. www.eia.gov/outlooks/aeo/electricity/sub-topic-03.php.
[3] X. Chen, Y. Du, E. Lim, H. Wen, K. Yan, J. Kirtley, Power ramp-rates of utility-scale pv systems under passing clouds: module-level emulation with cloud shadow modeling, Appl. Energy 268 (2020) 114980.
[4] K. Lappalainen, S. Valkealahti, Output power variation of different pv array configurations during irradiance transitions caused by moving clouds, Appl. Energy 190 (2017) 902—910.
[5] F. Crespi, A. Toscani, P. Zani, D. Sánchez, G. Manzolini, Effect of passing clouds on the dynamic performance of a csp tower receiver with molten salt heat storage, Appl. Energy 229 (2018) 224—235.
[6] C. Furlan, A.P. de Oliveira, J. Soares, G. Codato, J.F. Escobedo, The role of clouds in improving the regression model for hourly values of diffuse solar radiation, Appl. Energy 92 (2012) 240—254.
[7] M.J. De Smith, M.F. Goodchild, P. Longley, Geospatial Analysis: a Comprehensive Guide to Principles, Techniques and Software Tools, Troubador publishing ltd, 2007.
[8] W.J. Randel, F. Wu, D.J. Gaffen, Interannual variability of the tropical tropopause derived from radiosonde data and ncep reanalyses, J. Geophys. Res.: Atmosphere 105 (D12) (2000) 15509—15523.
[9] R.A. Houze Jr., Cloud Dynamics, Academic press, 2014.
[10] C. Deng, Z. Li, W. Wang, S. Wang, L. Tang, A.C. Bovik, Cloud detection in satellite images based on natural scene statistics and gabor features, Geosci. Rem. Sens. Lett. IEEE 16 (4) (2019) 608—612.
[11] T. Hong, P. Pinson, Y. Wang, R. Weron, D. Yang, H. Zareipour, Energy forecasting: a review and outlook, IEEE Open Access J. Power Energy 7 (2020) 376—388, https://doi.org/10.1109/OAJPE.2020.3029979.
[12] R. Perez, S. Kivalov, J. Schlemmer, K. Hemker, D. Renné, T.E. Hoff, Validation of short and medium term operational solar radiation forecasts in the us, Sol. Energy 84 (12) (2010) 2161—2172.
[13] M. Bouzerdoum, A. Mellit, A. Massi Pavan, A hybrid model (sarima—svm) for short-term power forecasting of a small-scale grid-connected photovoltaic plant, Sol. Energy 98 (2013) 226—235.
[14] A. Mellit, A. Massi Pavan, V. Lughi, Short-term forecasting of power production in a large-scale photovoltaic plant, Sol. Energy 105 (2014) 401—413.
[15] S. Salcedo-Sanz, C. Casanova-Mateo, A. Pastor-Sánchez, M. Sánchez-Girón, Daily global solar radiation prediction based on a hybrid coral reefs optimization — extreme learning machine approach, Sol. Energy 105 (2014) 91—98.
[16] P. Lauret, C. Voyant, T. Soubdhan, M. David, P. Poggi, A benchmarking of machine learning techniques for solar radiation forecasting in an insular context, Sol. Energy 112 (2015) 446—457.
[17] O. García-Hinde, G. Terrén-Serrano, M. Hombrados-Herrera, V. Gómez-Verdejo, S. Jiménez-Fernández, C. Casanova-Mateo, J. Sanz-Justo, M. Martínez-Ramón, S. Salcedo-Sanz, Evaluation of dimensionality reduction methods applied to numerical weather models for solar radiation forecasting, Eng. Appl. Artif. Intell. 69 (2018) 157—167.
[18] H. Jiang, N. Lu, G. Huang, L. Yao, J. Qin, H. Liu, Spatial scale effects on retrieval accuracy of surface solar radiation using satellite data, Appl. Energy 270 (2020) 115178.
[19] A.A. Prasad, R.A. Taylor, M. Kay, Assessment of direct normal irradiance and cloud connections using satellite data over Australia, Appl. Energy 143 (2015) 301—311.
[20] P. Mathiesen, C. Collier, J. Kleissl, A high-resolution, cloud-assimilating numerical weather prediction model for solar irradiance forecasting, Sol. Energy 92 (2013) 47—61.
[21] S. ichi Inage, Development of an advection model for solar forecasting based on ground data first report: development and verification of a fundamental model, Sol. Energy 153 (2017) 414—434.
[22] M. Diagne, M. David, P. Lauret, J. Boland, N. Schmutz, Review of solar irradiance forecasting methods and a proposition for small-scale insular grids, Renew. Sustain. Energy Rev. 27 (2013) 65—76.
[23] S. Sobri, S. Koohi-Kamali, N.A. Rahim, Solar photovoltaic generation forecasting methods: a review, Energy Convers. Manag. 156 (2018) 459—497.
[24] A.R. Pazikadin, D. Rifai, K. Ali, M.Z. Malik, A.N. Abdalla, M.A. Faraj, Solar irradiance measurement instrumentation and power solar generation forecasting based on artificial neural networks (ann): a review of five years research trend, Sci. Total Environ. 715 (2020) 136860.
[25] C.-L. Fu, H.-Y. Cheng, Predicting solar irradiance with all-sky image features via regression, Sol. Energy 97 (2013) 537—550.
[26] C. Shi, Y. Zhou, B. Qiu, J. He, M. Ding, S. Wei, Diurnal and nocturnal cloud

segmentation of all-sky imager (asi) images using enhancement fully convolutional networks, Atmos. Meas. Tech. 12 (2019) 4713–4724.

[27] C.W. Chow, B. Urquhart, M. Lave, A. Dominguez, J. Kleissl, J. Shields, B. Washom, Intra-hour forecasting with a total sky imager at the uc san diego solar energy testbed, Sol. Energy 85 (11) (2011) 2881–2893.

[28] S. Dev, Y.H. Lee, S. Winkler, Color-based segmentation of sky/cloud images from ground-based cameras, IEEE J. Sel. Top. Appl. Earth Obs. Rem. Sens. 10 (1) (2017) 231–242.

[29] H. Li, F. Wang, H. Ren, H. Sun, C. Liu, B. Wang, J. Lu, Z. Zhen, X. Liu, Cloud identification model for sky images based on otsu, in: International Conference on Renewable Power Generation, RPG 2015), 2015, pp. 1–5.

[30] L. Ye, Z. Cao, Y. Xiao, Z. Yang, Supervised fine-grained cloud detection and recognition in whole-sky images, IEEE Trans. Geosci. Rem. Sens. 57 (10) (2019) 7972–7985.

[31] D. Yang, J. Kleissl, C.A. Gueymard, H.T. Pedro, C.F. Coimbra, History and trends in solar irradiance and pv power forecasting: a preliminary assessment and review using text mining, Sol. Energy 168 (2018) 60–101.

[32] M. Gohari, B. Urquhart, H. Yang, B. Kurtz, D. Nguyen, C. Chow, M. Ghonima, J. Kleissl, Comparison of solar power output forecasting performance of the total sky imager and the university of California, san diego sky imager, Energy Procedia 49 (2014) 2340–2350, proceedings of the SolarPACES 2013 International Conference.

[33] R. Marquez, C.F. Coimbra, Intra-hour dni forecasting based on cloud tracking image analysis, Sol. Energy 91 (2013) 327–336.

[34] W. Kong, Y. Jia, Z.Y. Dong, K. Meng, S. Chai, Hybrid approaches based on deep whole-sky-image learning to photovoltaic generation forecasting, Appl. Energy 280 (2020) 115875.

[35] H.-Y. Cheng, C.-L. Lin, Cloud detection in all-sky images via multi-scale neighborhood features and multiple supervised learning techniques, Atmos. Meas. Tech. 10 (1) (2017) 199–208.

[36] B.J. Redman, J.A. Shaw, P.W. Nugent, R.T. Clark, S. Piazzolla, Reflective all-sky thermal infrared cloud imager, Opt Express 26 (9) (2018) 11276–11283.

[37] A. Mammoli, G. Terren-Serrano, A. Menicucci, T.P. Caudell, M. Martínez-Ramón, An experimental method to merge far-field images from multiple longwave infrared sensors for short-term solar forecasting, Sol. Energy 187 (2019) 254–260.

[38] H. Escrig, F. Batlles, J. Alonso-Montesinos, F. Baena, J. Bosch, I. Salbidegoitia, J. Burgaleta, Cloud detection, classification and motion estimation using geostationary satellite imagery for cloud cover forecast, Energy 55.

[39] J.A. Shaw, P.W. Nugent, N.J. Pust, B. Thurairajah, K. Mizutani, Radiometric cloud imaging with an uncooled microbolometer thermal infrared camera, Opt Express 13 (15) (2005) 5807–5817.

[40] J.A. Shaw, P.W. Nugent, Physics principles in radiometric infrared imaging of clouds in the atmosphere, Eur. J. Phys. 34 (6) (2013). S111–S121.

[41] B. Thurairajah, J.A. Shaw, Cloud statistics measured with the infrared cloud imager, IEEE Trans. Geosci. Rem. Sens. 43 (9) (2005) 2000–2007.

[42] P.W. Nugent, J.A. Shaw, S. Piazzolla, Infrared cloud imaging in support of earth-space optical communication, Opt Express 17 (10) (2009) 7862–7872.

[43] N.J.P. Paul, W. Nugent, Joseph A. Shaw, Correcting for focal-plane-array temperature dependence in microbolometer infrared cameras lacking thermal stabilization, Opt. Eng. 52 (6) (2013) 1–8.

[44] X. Hu, Y. Wang, J. Shan, Automatic recognition of cloud images by using visual saliency features, Geosci. Rem. Sens. Lett. IEEE 12 (8) (2015) 1760–1764.

[45] C. Shi, Y. Wang, C. Wang, B. Xiao, Ground-based cloud detection using graph model built upon super pixels, Geosci. Rem. Sens. Lett. IEEE 14 (5) (2017) 719–723.

[46] E. Ising, Beitrag zur Theorie des Ferromagnetismus, Zeitschrift fur Physik 31 (1) (1925) 253–258.

[47] L. Onsager, Crystal statistics. i. a two-dimensional model with an order-disorder transition, Phys. Rev. 65 (1944) 117–149.

[48] J. Besag, On the statistical analysis of dirty pictures, J. Roy. Stat. Soc. B 48 (3) (1986) 48–259.

[49] C. Papin, P. Bouthemy, G. Rochard, Unsupervised segmentation of low clouds from infrared meteosat images based on a contextual spatio-temporal labeling approach, IEEE Trans. Geosci. Rem. Sens. 40 (1) (2002) 104–114.

[50] Q. Li, W. Lyu, J. Yang, J. Wang, Thin cloud detection of all-sky images using markov random fields, Geosci. Rem. Sens. Lett. IEEE 9 (2012) 417–421.

[51] S. Liu, L. Zhang, Z. Zhang, C. Wang, B. Xiao, Automatic cloud detection for all-sky images using superpixel segmentation, Geosci. Rem. Sens. Lett. IEEE 12 (2)

[52] A. Taravat, F. Del Frate, C. Cornaro, S. Vergari, Neural networks and support vector machine algorithms for automatic cloud classification of whole-sky ground-based images, Geosci. Rem. Sens. Lett. IEEE 12.

[53] W. Zhuo, Z. Cao, Y. Xiao, Cloud classification of ground-based images using texture–structure features, J. Atmos. Ocean. Technol. 31 (1) (2014) 79–92.

[54] J. Drönner, N. Korfhage, S. Egli, M. Mühling, B. Thies, J. Bendix, B. Freisleben, B. Seeger, Fast cloud segmentation using convolutional neural networks, Rem. Sens. 10 (2018) 1782.

[55] J. Zhang, L. Pu, F. Zhang, Q. Song, Cloudnet: ground-based cloud classification with deep convolutional neural network, Geophys. Res. Lett. 45 (16) (2018) 8650–8672.

[56] G. Terrén-Serrano, M. Martínez-Ramón, Processing of Global Solar Irradiance and Ground-Based Infrared Sky Images for Very Short-Term Solar Forecasting, 2021 arXiv:2101.08694.

[57] J. Shawe-Taylor, N. Cristianini, Kernel Methods for Pattern Analysis, Cambridge University Press, New York, NY, USA, 2004.

[58] C.J. Burges, A tutorial on support vector machines for pattern recognition, Data Min. Knowl. Discov. 2 (2) (1998) 121–167.

[59] C. Williams, C.E. Rasmussen, Gaussian Processes for Machine Learning, vol. 2, MIT press Cambridge, MA, 2006.

[60] G. Terrén-Serrano, A. Bashir, T. Estrada, M. Martínez-Ramón, Girasol, a sky imaging and global solar irradiance dataset, Data in Brief (2021) 106914.

[61] S. Hess, Introduction to Theoretical Meteorology, Holt-Dryden book, Holt, 1959.

[62] P.H. Stone, J.H. Carlson, Atmospheric lapse rate regimes and their parameterization, J. Atmos. Sci. 36 (3) (1979) 415–423.

[63] J. Hummel, W. Kuhn, Comparison of radiative-convective models with constant and pressure-dependent lapse rates, Tellus 33 (3) (1981) 254–261.

[64] L. L. Pan, L. A. Munchak, Relationship of cloud top to the tropopause and jet structure from calipso data, J. Geophys. Res.: Atmosphere 116 (D12).

[65] G. Terrén-Serrano, M. Martínez-Ramón, Multi-layer wind velocity field visualization in infrared images of clouds for solar irradiance forecasting, Appl. Energy 288 (2021) 116656.

[66] S. Baker, R. Gross, T. Ishikawa, I. Matthews, Lucas-kanade 20 years on: a unifying framework: Part 2, Int. J. Comput. Vis. 56 (2003) 221–255.

[67] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, Springer Series in Statistics, Springer New York Inc., 2001.

[68] K.P. Murphy, Machine Learning: a Probabilistic Perspective, MIT press, 2012.

[69] E. Forgy, Cluster analysis of multivariate data: efficiency versus interpretability of classification, Biometrics 21 (3) (1965) 768–769.

[70] S.Z. Li, Markov Random Field Modeling in Image Analysis, Springer-Verlag, Berlin, Heidelberg, 2001.

[71] D. Kolle, N. Friedman. Probabilistic Graphical Models: Principles and Techniques, MIT Press, 2009.

[72] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, Science 220 (4598) (1983) 671–680.

[73] Z. Kato, T.-C. Pong, A markov random field image segmentation model using combined color and texture features, in: W. Skarbek (Ed.), Computer Analysis of Images and Patterns, Springer Berlin Heidelberg, 2001, pp. 547–554.

[74] M.A. Aizerman, E.M. Braverman, L. Rozoner, Theoretical foundations of the potential function method in pattern recognition learning, Autom. Rem. Contr. 25 (1964) 821–837.

[75] B. Schölkopf, R. Herbrich, A.J. Smola, A generalized representer theorem, in: International Conference on Computational Learning Theory, Springer, 2001, pp. 416–426.

[76] A. Navia-Vazquez, F. Pérez-Cruz, A. Artes-Rodriguez, A.R. Figueiras-Vidal, Weighted least squares training of support vector classifiers leading to compact and adaptive schemes, IEEE Trans. Neural Network. 12 (5) (2001) 1047–1059.

[77] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, C.-J. Lin, Liblinear, A library for large linear classification, J. Mach. Learn. Res. 9 (2008) 1871–1874.

[78] C.-W. Hsu, C.-C. Chang, C.-J. Lin, A Practical Guide to Support Vector Classification, 2010.

[79] T. Jaakkola, M. Jordan, A Variational Approach to Bayesian Logistic Regression Models and Their Extensions, 1997.

[80] W.J. Youden, Index for rating diagnostic tests, Cancer 3 (1) (1950) 32–35.

[81] T. Fawcett, An introduction to roc analysis, Pattern Recogn. Lett. 27 (8) (2006) 861–874.