Informatik-Bericht Nr. 2012-2

Schriftenreihe Fachbereich Informatik, Fachhochschule Trier

Detection of Bird Activity in Radar Images

Yannick Thesen, Peter Gemmar

Trier, April 2011

Abstract. Bird strikes pose a serious risk to aircraft, resulting in significant economical losses due to damages and aircraft downtime. As up to 90% of birds strikes occur during landing or taking off, bird control is an important airport safety matter. In order to assess the abundance of birds in the vicinity of the airport, observation over an extended time period is required. Marine surveillance radars, when operated in vertical mode (with the antenna tilted 90° upward), have proven effective in quantifying bird activity up to an altitude of several kilometers and are operable night and day, even in poor visibility conditions. In a study conducted in the years 2007-2008, such a radar system was deployed at the airport of Leipzig/Halle (LEJ), Germany. During this study, screenshots of the radar's plan position indicator (PPI) display were acquired once per minute using a frame grabber. In the near future a similar study will be conducted at Munich airport (MUC), using a higher capturing rate.

The objective of this project was to create a software application for automatic classification and quantification of birds in radar images acquired as described above, with the intention of using it in the upcoming study in Munich. A software prototype was developed which incorporates methods to preprocess the images, dynamically detect ground clutter interfering with target detection and detect contaminations caused by insects and precipitation. Additionally, various features of detected echoes can be calculated and inspected using scatter plots, principal component analysis and self-organizing maps. Support vector machines can be trained using a training set labeled using the prototype.

Unfortunately, no verified "ground truth" training set was available while writing this thesis, rendering the design of a final classifier more complicated. Nevertheless, a multi-level classifier architecture based on the findings of this project and the solutions implemented in the prototype is proposed. Additional investigations suggest, that the practice of discarding weak radar echoes is most likely wrong, as echo appearance is sensitive to the target's position and aspect in the radar beam. This is confirmed by inspection of images captured at higher rates. To prevent underestimations of bird activity because objects are missed or echoes are weak in the moment of capturing, the potentials of images captured at higher rates (e.g. target tracking) should be exploited.

Contents

1

Introduction

There are various motivations for quantifying large-scale bird migration as well as low-scale bird movements. Besides the interest of the scientific community, namely ornithologists and biologists, there is an increased need for environmental impact studies which assess the risk of collisions between birds and artificial structures and the effects of these structures on birds. One of the most prominent and most recent examples of such human-wildlife conflicts is the ditching of US Airways flight 1549 in the Hudson River in New York on January 15, 2009, following the ingestion of at least one Canada goose in each engine shortly after the plane's departure from LaGuardia Airport [1]. Fortunately, all 155 passengers and crew were safely evacuated from the aircraft. The Airbus however, was damaged beyond repair.

Although a long-known fact among experts in the aviation industry, the intense media coverage accompanying this incident brought world-wide attention to the fact that birds pose a substantial risk to aircraft, causing thousands of bird strikes per year. As approximately 90% of all bird strikes occur close to airfields [2, 3], assessing the abundance and behavior of migratory and non-migratory birds in the vicinity of airports is an important part of bird strike management plans.

Further need for the quantification of bird activity comes with increasing investments in wind energy. With a growing environmental awareness and increased demand for sustainable energy sources, much effort is put into the construction of both onshore and offshore wind farms. Bird migration monitoring is an integral part of environmental impact studies conducted in the planning stage of wind farms. In the post-construction phase, quantifying the amount of accidental collisions and habitat loss as well as determining if wind farms pose a barrier to migration routes or general freedom of movement is of great interest [4, 5].

Traditional visual observation of birds movements is obviously a very time-consuming process and it is difficult to monitor birds migrating at high altitudes or during nighttime. When it comes to offshore wind farms, these problems become even more prominent, with the additional logistical efforts required for effective monitoring in the open sea. Radar technology is therefore one of the most important tools for bird observation and its use is well-documented in the literature (e.g. [6, 7, 8, 9, 10]). Unlike traditional techniques, radar observation is independent from visual conditions such as darkness or foggy weather and can be performed autonomously during long time periods. Radar observation covers

large volumes, yielding more accurate bird altitude information, flight directions and flight speeds. Besides the use of weather radar data for the quantification of large-scale bird migration (e.g. [11]), much effort is directed at the utilization of commercial off-the-shelf (COTS) marine radars for low-scale monitoring or the customization of such radars to meet the specific requirements of radar ornithology. This has resulted in the development of several operational systems, some of which are commercially available [12, 13, 14].

Bird radars produce significant amounts of data that is virtually impossible to analyze by hand. The objective of this project was the development of a software application for processing and interpreting image data obtained using a marine radar deployed at airports with the aim of quantifying bird activity in the airport vicinity in relation to the time of day and year. The data used in this project was acquired at the airport of Leipzig/Halle (LEJ), Germany, with the intention of applying the developed methods to images which will be captured in a future study at the airport of Munich (MUC).

After an introduction to bird strike management, radar technology and a short review of existing systems and approaches, the methods developed within the scope of this project are described (chapter 2) and evaluated (chapter 3).

1.1 Bird Strike Management

Bird strikes pose a serious risk to aircraft, resulting in significant economical losses. In the years 2007 and 2008, 2218 bird strikes involving civil German aircraft were recorded. This resulted in damages in 98 cases, the most prominent consequences of which were precautionary landing, aborted take-offs and engine cutoff [3]. In general, there exists an upward tendency regarding bird strike incidents. In the United States, for example, about 7,670 wildlife strikes with civil aircraft were reported in 2007, compared to 1,759 in 1990, while commercial aircraft movements increased only about 13% during this period [1, 15].

It is estimated that, on a worldwide scale, civil aviation suffered an estimated number of 54 fatal accidents involving birds in the period of 1912 to 2008, killing 276 people and destroying 108 aircraft [16]. In military aviation, more than 380 serious accidents resulting in damages beyond repair and loss of human lives were reported for 1950-2002 [17]. In general, the actual number of bird strikes both for civil and military aviation is possibly much higher, as records are incomplete (it is expected, that pilots report only 20-30% of bird strikes [2]) and limited information is available for many regions of the world.

Besides the direct damage to aircraft and substantial risks to human life, bird strikes are responsible for major economic losses due to aircraft downtime. According to [15], wildlife strikes cost the civil aviation industry an estimated 118,448 hours per year of aircraft downtime and $123 million in monetary losses in the United States alone.

It has been shown, that up to 90% of reported bird strikes occur in the vicinity of airfields, inside of the 13-km "Bird Circle" defined by the International Bird

Strike Committee (IBSC) [2, 3] and below an altitude of 500ft [18]. Bird control plans are therefore an important airport safety matter. Reasons for an increased bird strike hazard at airfields include adaption of birds to urban environments and the attractiveness of large areas of grass and pavement for feeding and resting, as well as modern aircraft becoming quieter and therefore less obvious to birds. Additionally, efforts to protect the environment over the last 40 years resulted in an increase in bird populations, which can be as dramatic as is the case with the population of Canada geese in North America, which has more than quadrupled from 1.2 to 5.5 million in the time between 1970 and 2008 [1].

In the following, the two basic elements of bird control plans, *habitat management* and *active bird control*, will be introduced, as they form the motivation for avian radar observation at airports.

1.1.1 Airfield Habitat Management

There are two basic challenges in airfield habitat management: (1) Identifying attractive features of an airfield and (2) removing the attraction. Airport infrastructure potentially contains various features attractive to birds. One example of such infrastructure are devices close to the runway (e.g. antennas, 1000ft markers, fences, runway and landing lights) which birds perch on. Paved areas such as runways, ramps and service areas are attractive loafing sites. Wetlands and other natural water bodies as well as artificial basins and water accumulating on hard surfaces can attract birds and especially waterfowl, which are of increased concern because of their size and flocking behavior. Woody vegetation such as trees, shrubs and hedgerows are vegetative habitat while also providing cover for deer, which is the greatest hazard regarding potential aircraft damage. Tall vegetation is attractive to large ground-nesting birds and supports large populations of prey such as the invertebrate and small mammals. Finally, agricultural production close to airfields significantly increases birds strike risk and should be contained to areas far away from the runways (recommendations regarding a minimum distance range from 170m to 1.2km) [18, 19].

There are various typical counter measures taken to reduce the attractiveness of the airfield, which usually consist of reducing (access to) food, water, loafing sites and shelter. Water sources should be removed or access prevented by wire-grid systems or floating plastic balls while water accumulation on surfaces can be reduced by constructing graded surfaces and taking drainage measures. It is debatable whether keeping vegetation short or tall is of significant use, however, choice of appropriate vegetative species may help keeping attractiveness low and woody vegetation should without question be removed [18]. Additionally, access to perching sites can be reduced using porcupine wire[1], monofilament nylon lines or sticky substances. It has been suggested by [19], that there is significant room for improvements in habitat management by making further changes to hard infrastructure (e.g. worm-proof drainage gutters) and by making it a requirement

---

[1] Porcupine wire is a type of barbed wire consisting of numerous stainless steel spikes, which is used to prevent birds and other wildlife from sitting or climbing on buildings and other surfaces.

for manufacturers of devices in the runway environment to include anti-perching functionality in the design of their equipment.

In October 2006, the International Birdstrike Committee released a set of standards, that "should apply to any aerodrome carrying regular scheduled commercial air traffic, irrespective of the movement frequency or type of aircraft involved" [20]. Standard 2 of the document stresses the requirement of habitat management:

> An airport should undertake a review of the features on its property that attract hazardous birds/wildlife. The precise nature of the resource that they are attracted to should be identified and a management plan developed to eliminate or reduce the quantity of that resource, or to deny birds access to it as far as is practicable.
> Where necessary, support from a professional bird/wildlife strike prevention specialist should be sought.
> Documentary evidence of this process, its implementation and outcomes should be kept.

Standard 9 further elaborates on risks assessments and the requirement to determine what management processes should be implemented:

> Airports should conduct an inventory of bird attracting sites within the ICAO defined 13km bird circle, paying particular attention to sites close to the airfield and the approach and departure corridors. A basic risk assessment should be carried out to determine whether the movement patterns of birds/wildlife attracted to these sites means that they cause, or may cause, a risk to air traffic. If this is the case, options for bird management at the site(s) concerned should be developed and a more detailed risk assessment performed to determine if it is possible and/or cost effective to implement management processes at the site(s) concerned. This process should be repeated annually to identify new sites or changes in the risk levels produced by existing sites. (...)

1.1.2 Active Bird Control

Besides habitat management measures, it is necessary to actively monitor and control bird strike hazards during normal airport operation, so that even small numbers of hazardous birds and/or wildlife can be dispersed from the airfield, preventing them from attracting further individuals and so minimizing the risk of a bird strike. This requires the use of a mobile patrol consisting of trained and equipped bird controllers, which is dedicated to dispersing birds immediately after their detection and which is present whenever an aircraft lands or takes off. Bird deterrent measures include usage of visual devices (e.g. scare crows, hand held lasers), acoustic devices (e.g. radio controlled sound generators, pyrotechnic pistols, distress call appliances), lethal devices (traps, shotguns) and the deployment of trained predators, such as falcons, dogs or even radio-controlled bird models. A major problem is that static devices gradually lose their effectiveness with birds becoming habituated, so that portable devices are of great importance [20].

Monitoring bird management programs is essential to assess the effectiveness of the applied measures and to identify how programs may be extended or improved. Whenever bird movements need to be reliably quantified over extensive time periods, remote sensing technology such as radar is essential. The following section will give a short introduction to the principles of radar sensing, followed by a review of existing approaches and applications.

1.2 Remote Sensing using Radar

Radar ("Radio Detection and Ranging") is a remote sensing technology which is free of some basic drawbacks of other sensing utilities such as telescopes or photo and video cameras. (1) Radar is an active system which doesn't depend on an external source of illumination and is thus able to operate night and day. (2) as the radar beam uses the electromagnetic spectrum for sensing, it is not as impeded by fog, rain and snow as is the case with visual observation. One disadvantage of radar is its limited resolution, rendering target details mostly indiscernible. Radar as an active, autonomous sensing system incorporates detection, localization, state of motion assessment, classification and identification of objects or *targets*. While military applications dominated initial advances in radar technology, after World War II, radar has been increasingly used in civil applications such as air traffic control, weather reconnaissance, remote sensing of the earth's surface and speed cameras [21].

Radar systems can be categorized into (1) surveillance radars, which are designed for continuously searching a large volume (e.g. weather radar and air surveillance radar), (2) tracking radars, which are able to track single targets (e.g. military fire-control radar), (3) instrumentation radars measuring distances, velocities or flight paths and (4) synthetic-aperture radars (SAR), used for remote sensing of the earth's surface from satellites or aircraft [21]. When observing birds, a tracking radar offers the advantage, that individual bird targets can be monitored as long as they are in range of the radar, possibly allowing more detailed classification of the bird. Its key disadvantage is that it cannot deliver a quantification of bird movements, which a surveillance radar is capable of. Although tracking radars have been used in bird ornithology applications (e.g. [11, 22, 23]), the radar that provided the images used in this project is a standard marine surveillance radar.

The basic principle of radar is the transmission of electromagnetic wave pulses with a certain power P_t and an antenna gain of G_t in the direction of some potential target. Assuming that the target has a distance of R from the radar, the power density at the target position (a point on the surface of a sphere with radius R) is

$$P_{target} = \frac{P_t \cdot G_t}{4\pi \cdot R^2} \tag{1.1}$$

The amount of energy reflected depends on the radar cross-section (RCS) of the target, σ, which is basically its reflecting area. Assuming that the beam is reflected equally in all spatial directions and that the radar antenna has an effective surface (aperture) A_r, the energy received by the radar is[2]

$$P_{radar} = \frac{P_t \cdot G_t}{4\pi \cdot R^2} \cdot \frac{\sigma}{4\pi} \cdot \frac{1}{R^2} \cdot A_r \tag{1.2}$$

$$= \frac{P_t \cdot G_t \cdot \sigma \cdot A_r}{(4\pi)^2 \cdot R^4} \tag{1.3}$$

---

[2] All formulae in this section assume a hypothetical radar positioned in empty space. One could include additional terms $L_1 \cdot L_2 \cdot \ldots \cdot L_N$ in the formula denominators to take sources of energy loss such as atmospheric attenuation into account.

Equation 1.3 illustrates, that there is a $\frac{1}{R^4}$ dependency of the received energy on the target distance. The radar cross-section of a target depends on the signal frequency and on the aspect of the target [21].

In order to suppress the detection of small objects such as insects or sea waves, most radar devices make use of a sensitivity time control (STC) filter which reduces the sensitivity of the receiver with decreasing distance, thus excluding targets with cross-sections below a certain size. The influence of the STC is rarely documented when it comes to marine radars, which makes assessing the effects of different STC-settings on detection rates difficult. Obviously, the STC-setting must not be changed within an observation period as doing so will render the observations incomparable [24].

A major issue for radar antennas is extraneous side-lobe energy emitted from the radar (see figure 1.1) and main lobes striking the ground causing ground clutter echoes that can interfere with target detection. Depending on the antenna type and the radar surroundings (e.g. airport infrastructure or vegetation), these interferences can render the signals received from large portions of the observed volume difficult or impossible to analyze [25].
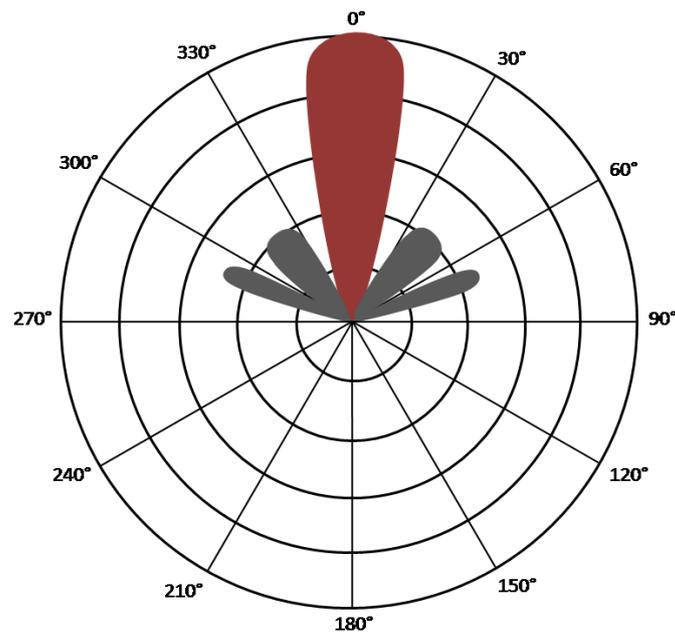

Fig. 1.1. Side lobes of radar antenna (sketch). Main lobe is shown in red, side lobes are shown in grey.

When making use of radar technology, various technical aspects need to be considered. A major factor is the choice of antenna, which, for avian radars, is usually made between the slotted array antenna and the parabolic dish antenna. Slotted array antennas, when operated in horizontal mode, are useful for obtaining detailed range and bearing information but cannot provide altitude discrimination.

The detection volume (see figure 1.2) depends on the beam shape, which is usually 1-2 degrees wide and 10 or more degrees up and down from horizontal.
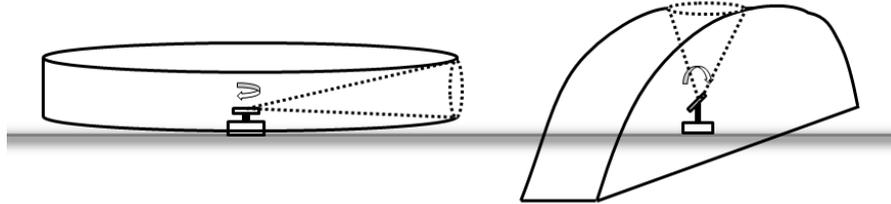

Fig. 1.2. Schematic illustration depicting the observed volume of a radar equipped with an array antenna. Left: Default horizontal operation. Right: Vertical operation.

Parabolic dish antennas ("pencil-beam radar") on the other hand produce a defined conical beam (2-5 degrees wide) projected from the antenna, allowing target detection to be related to the known height of the beam at a given range (see figure 1.3). This provides an improved estimation of target altitude and allows radar systems using such an antenna to perform better regarding ground clutter formation, compared with horizontal operation of slotted array antennas [25].
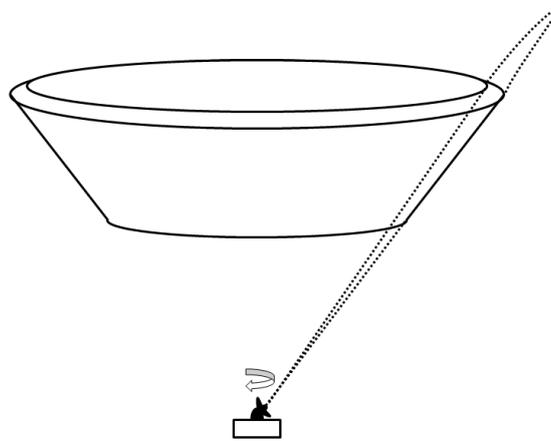

Fig. 1.3. Schematic illustration of a radar using a pencil beam antenna.

When quantitative altitude information is required, one option is to tilt a slotted array antenna 90 degrees towards the sky, so that the observed area becomes a fan-shaped volume that lies in a vertical plane (see figure 1.2). Such an antenna setup allows the height of targets crossing the radar lobe to be determined up to significantly larger altitudes compared to normal horizontal operation. However, tracking objects and thus determining target velocity in vertical operation is constrained by the fact that only a small horizontal area is covered by the radar and most birds stay in the beam for a rather short period.

Clearly, an ideal avian radar antenna should provide good elevation coverage, elevation accuracy and azimuth resolution. As no antenna on today's market can deliver such a performance (except at prohibitively large costs), it has been suggested in [26], that developing a suitable antenna is feasible and should be a top priority for parties interested in radar ornithology. However, using COTS radar systems remains a cost-efficient alternative.

1.3 Radar Setup

The radar system installed at the airport of Leipzig/Halle, which captured the data used in this project, is a Furuno FAR-2127BB, X-band (9410 $\pm$ 30 MHz), 25 kW marine radar with a 8ft array antenna operating in vertical mode. The system was positioned close to the end of the northern runway and perpendicular to the prevalent flyway (30°, 210°) of migrating birds so that as many birds as possible may be detected. The PPI (plan position indicator) output of the radar system was captured at regular intervals using a frame grabber (DVI2USB, Epiphan) connected via DVI (Digital Visual Interface). The system was configured to cover altitudes up to 1.5 km with the radar gain set to maximum (STC off). To keep the resulting data volume manageable, the frame grabber was configured to capture one image per minute which was saved as a PNG (Portable Network Graphics) file using a laptop computer [27].

While setting the capturing frequency of the frame grabber to one image per minute reduces memory requirements, it does have some drawbacks when it comes to evaluating the data. Because of the great temporal distance between subsequent images, target tracking is rarely possible and it is likely that significant numbers of birds will be missed if they are flying at high speeds and at steep angles to the disk-shaped volume observed by the radar. However, especially with weak echoes, incorporating information from the neighboring images can provide valuable information regarding the nature of the echo. A weak echo will often become stronger in subsequent images when the target is just entering the volume observed by the radar and conversely, a strong echo may faint with time when leaving the observed volume (see figure 1.5). Obviously, with images captured at one minute intervals, matching echoes originating from the same target in subsequent images is not possible. Thus, birds may either be counted multiple times (if staying in the observed area long enough to be redetected) or echoes might be dismissed as too weak although it would be correct to include them in the bird statistics. For these reasons, the upcoming study of bird activity at Munich airport will be performed with a higher capturing frequency.

Unfortunately, the only images that have been available during the development of the methods described in this project were acquired using the low capturing rate. Therefore, all of the developed methods work on single images, not incorporating target information derived from neighboring frames. They are, however, equally applicable to images acquired at a higher frequency and are easily extendable when new data becomes available.
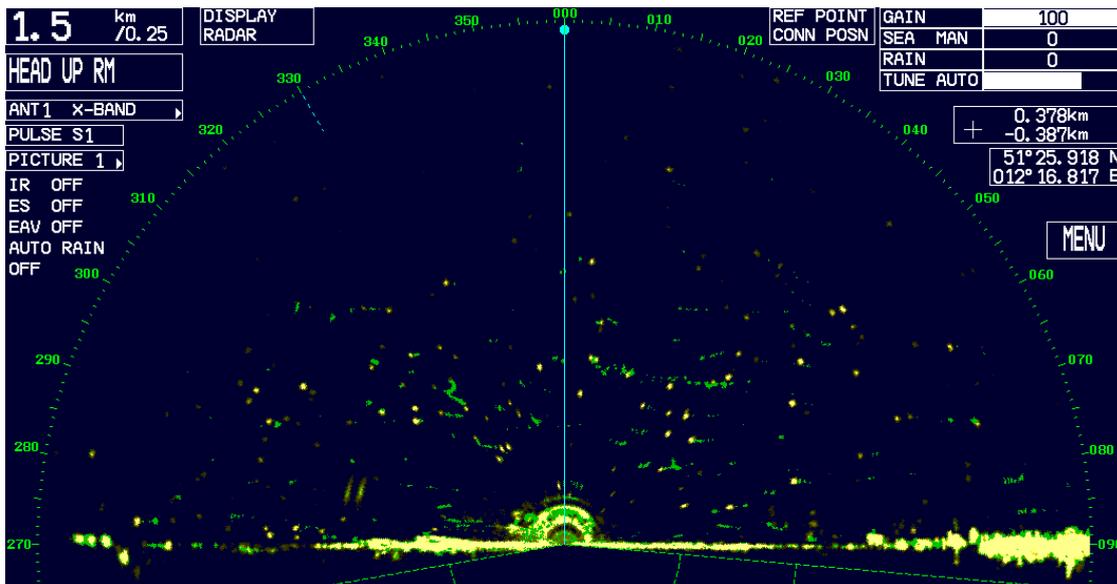
(a) Image with numerous bird echoes.
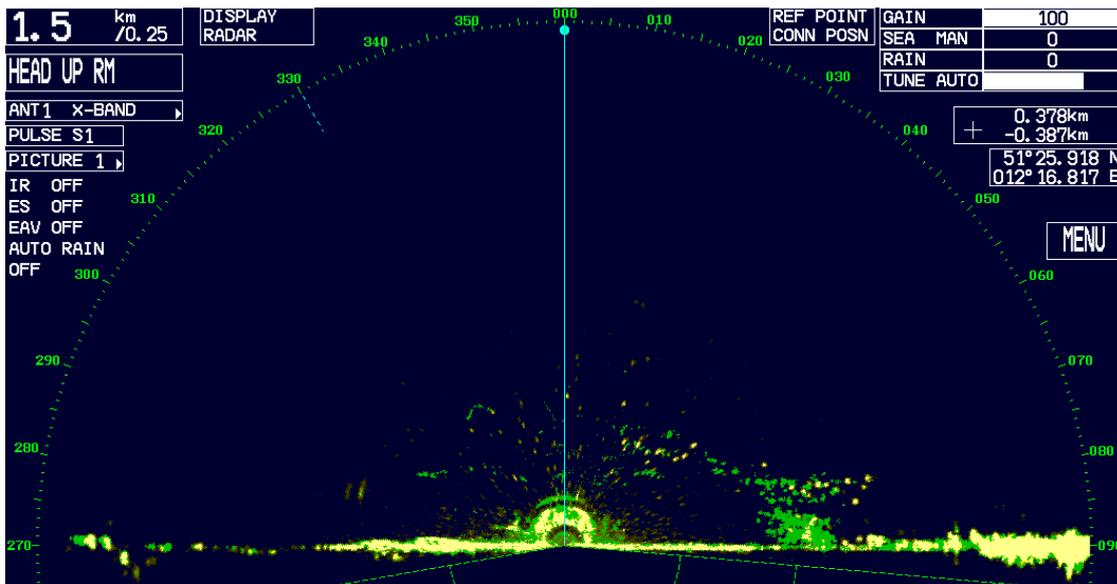
(b) Image with insect echoes (low intensity pixels) at the center and a bird flock right of the radar position.

Fig. 1.4. Screenshots of radar PPI display depicting typical situations. Current echoes are colored yellow, trails accumulated over the last minute are colored green. (Showing an extract of the PPI display. See figure 2.1 for a full screenshot.)

Figure 1.4 shows two images captured as described above. Both images contain significant amounts of ground clutter, to be observed in the lower part of the images. In these areas, bird detection is not possible. Image 1.4(a) shows a situation where numerous bird echoes have been recorded without any interferences being present (except for ground clutter). In this image, most of the echoes correspond to birds and interpretation is rather straight forward. The image in figure 1.4(b) shows echoes of birds flying in formation in the bottom right part of the screen, as well as some contamination caused by insects in the vicinity of the radar. Given the varying appearance of bird echoes depending on bird size, flocking behavior and aspect in the radar beam, as well as the possibility of insect contamination, there are several scenarios that an automatic classification system needs to be able to interpret.

1.4 Preliminary Work and Data Basis

In the course of the study assessing bird movements at the airport of Leipzig/Halle during which the radar images used in this project were captured, a software tool for the automated evaluation of these images has been created ("BirdWatch"). The program is able to discriminate between echoes corresponding to birds, airplanes and other objects with a sensitivity of at least 80% and a specificity of more than 84% [28, 27]. However, there are several aspects suggesting that the development of an improved classification system might be beneficial:

- The images based on which "BirdWatch" has been developed were captured at one minute intervals. As described above, this has some drawbacks when evaluating the data. As a higher capturing frequency reveals valuable information about echoes which can be used in interpreting the observed radar scene, developing a system which makes use of the possibilities of this higher resolution data seems appropriate.
- With a classification sensitivity (specificity) of 80% (84%), there is still a significant amount of misclassifications. In [27], automatic classification was followed by a manual review of the results with corrections performed when necessary. According to the author, this increased the classification rate to well over 90%. However, considering the huge amount of radar images which need to be verified by hand, improving the performance of the automatic classifier will significantly reduce the requirement for human interaction and thus speed up the evaluation of the data.
- While the ground clutter situation changes with time even if the radar system is not moved (e.g. because of seasonal changes in vegetation), "BirdWatch" uses a static and manually created clutter map. In order to maximize the number of detected birds (including those located close to clutter) and because clutter map generation is automatable, this step should be dynamically performed by the classification program.
- The "BirdWatch" software follows a strict regiment of rejecting images contaminated with insects, even if the contamination is limited to low altitudes or the

level of contamination is not exceedingly high. In most cases, detection of birds outside the affected region or in slightly contaminated regions is still feasible and should be performed by the evaluation software.

- In the course of this project, it became apparent, that the ground truth used in the training of the "BirdWatch" software might have been biased by using a computer monitor with limited resolution and suboptimal color display as well as by the assumption, that weak echoes should not be classified as birds. The former means that the expert providing the reference classifications in the training set might have been misled by the appearance of radar echoes on a certain monitor type. The latter indicates, that in the training set used in [28], weak echoes were consequently dismissed, although it can be assumed, that weak echoes are in fact often caused by birds which are on the point of leaving or entering the volume observed by the radar, or whose aspect in the moment of detection results in a low radar cross-section. Additionally, bird sizes vary up to four orders of magnitude (0.02-10kg) [10], resulting in very differently sized echoes. Figure 1.5 illustrates the changing appearance of bird targets when tracked in subsequent images captured at shorter intervals.
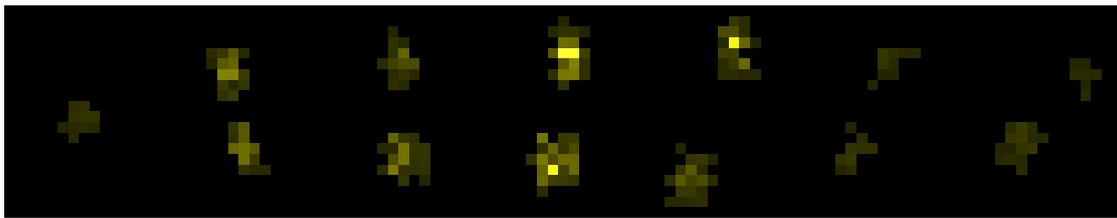
(a) Two bird targets detected in subsequent images.
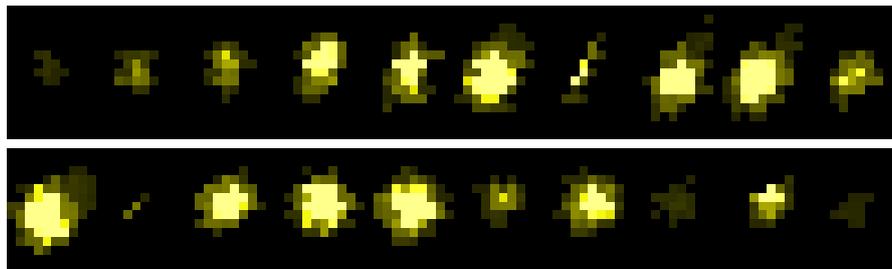
(b) Varying appearance of one single bird target with time. Ordered from top left to top right, then bottom left to bottom right.

Fig. 1.5. Varying appearance of bird targets in subsequent images captured at short intervals.

1.5 Literature Review

There are various publications on radar ornithology and specifically on using commercial off-the-shelf marine radars for quantifying bird movement.

In [25], Herricks describes the deployment of two customized avian radar systems by researchers of the University of Illinois Center of Excellence for Airport Technology (CEAT). The setup included three Furuno marine radars, two of which were outfitted with parabolic dish antennas set at +5 degrees. The third antenna used was an array antenna set at zero degrees. The radar data was collected 24/7 and analyzed using the Accipiter (see below) avian radar system.

[5] describes the deployment of a horizontal S-band radar and a vertical X-band radar, connected to a high-resolution radar capture card capable of recording echoes at $4,096$ levels of resolution, which is at least 256 times finer than standard output of a commercial radar. Based on the radar data, dynamic maps of background clutter are generated and birds are identified, tracked and categorized according to their size.

Researchers of the Swiss Ornithological Institute have since the 1960's been conducting studies using the "Superfledermaus", an ex-military fire-control radar operating in the X-band with a peak pulse power of 150kW and 2.2° nominal beam width (e.g. [29, 30, 8, 24, 22, 10]). In recent studies, the radar was alternately operated in fixed beam (quantitative) and tracking (qualitative) mode with simultaneous visual observation using a telescope mounted parallel to the radar antenna. Fixed beam measurements are performed with the radar being kept stationary pointing in a certain direction, while in tracking mode, single targets are detected and tracked as long as possible. The radar supplies echo signatures which can be used to classify birds and discriminate them from insects based on their distinct wing-beat pattern. In [23], Serge Zaugg et al. describe an automatic classification system for identifying bird targets based on the echo signatures obtained using the "Superfledermaus" (making use of wavelet transforms and support vector machines).

In [11], Dokter et al. showed the correlation between bird migration detected in operational weather radar and birds movements detected using the "Superfledermaus" radar at three sites in the Netherlands, Belgium and France. Dokter concludes, that bird migration detection in weather radar data is feasible (with detection probabilities of up to 99%) and that the developed methods could be used in a continent-wide bird migration sensor network based on operational weather radar [11, 22].

In [24], Heiko Schmaljohann provides a critical discussion of existing approaches, stating that in most cases, the detection probability of targets and the size of the volume surveyed by the radar remains unassessed, resulting in significant mistakes in bird density estimations. According to Schmaljohann, no single radar system on the market today can provide all the information required for proper quantification of bird movements: radar cross-section, echo signature, air speed, flight direction and position in the radar beam of every single target. Therefore, at least two observational approaches need to be combined when aiming to perform a precise

assessment. Schmaljohann further states, that calibration of the deployed radar is essential for determining the radar cross-sections of different targets at different distances, thus enabling the observed volume to be estimated.

One example of commercialized avian radar systems is the Merlin radar series developed by DeTect Inc. (Panama City, Florida, USA), which, besides bird detection, offers various features ranging from networking capabilities to automatic activation of bird deterrent devices. The system is usually equipped with one horizontally and one vertically scanning antenna and is able to detect small bird targets within a radius of about 3.2 km and up to an altitude of 1.5 km (large birds up to 4.8 km horizontally and 3.0 km vertically). Merlin systems include a software for bird detection, which is capable of removing interferences (such as weather or ground clutter) and which records numerous parameters for the detected targets. The company describes the software as the "most powerful bird and bat detection software on the market today" [12]. According to "TONI Vogelschlag-Prävention" ([31], 2010), the distributor of Merlin radars in Germany, about 45 Merlin systems have been deployed worldwide.

Accipiter is another manufacturer of commercial avian radars, the development of which is described e.g. in [13, 26, 7]. The radar allows for bird detection using adaptive clutter maps and real-time tracking.

2

Materials and Methods

This chapter describes the methods applied for clutter and insect/precipitation detection[1], target segmentation and feature calculation as well as evaluation of feature distributions. After describing these prerequisites, a system architecture for bird classification is proposed.

In [32], Awcock and Thomas proposed a generic model for machine vision systems, consisting of seven distinct modules. As many aspects of this model are applicable here, the following gives an overview of the model and links it to the methods presented in this project.

- *Scene constraints.* The observed scene must show those aspects which are required for automatic interpretation of the images and must be consistent in terms of properties such as lighting. Examples for such constraints when performing radar measurements are choosing a radar location that minimizes ground clutter, minimizing interferences by other equipment and using a consistent configuration of the radar system during the complete study.
- *Image acquisition.* This includes configuration and type of imaging device, so that the resulting images suffice the requirements of the application. In the context of radar ornithology, choice of radar hardware and antennas significantly affects the resulting image data (see section 1.2). An additional choice can be made between using the images rendered by the radar system and processing the raw radar data (which in case of COTS marine radars is often not possible without modifications to the radar hardware). In this project, the imaging device consisted of a frame grabber capturing the output of a marine radar PPI display. Configuration includes the capturing rate of the frame grabber (one image per minute) as well as range of the radar (1.5 km), gain (maximum), STC filter setting (off), rotation speed (24 rpm) and trail afterglow setting (on, length of one minute).
- *Image preprocessing.* Preprocessing ensures, that acquired images are free of noise, well framed and ready for further processing. Section 2.1 describes some preprocessing steps, including removal of background pixels and PPI display

---

[1] The applied methods for clutter and contamination map generation were largely developed by Tobias Dreimüller as part of his bachelor thesis written simultaneously with the present project.

graphical user interface (GUI) elements, as well as the definition of a region of interest (ROI).

- *Segmentation.* In the segmentation step, objects in the scene are separated from the background. For radar images obtained in vertical mode, this is rather straightforward because of the uniform radar screen background. Some additional processing is required in contaminated or cluttered areas and in case echoes of multiple targets have been merged by the radar system into one echo, which is then split by the software.

- *Feature extraction.* After object segmentation, features describing different (e.g. morphological) aspects of the detected objects are derived. Section 2.4.3 lists the object properties that are computed for the detected radar echoes and section 2.5 describes how features relevant to the application can be determined from the set of computed properties.

- *Classification/Interpretation.* Classification uses the previously extracted features to make decisions regarding the nature of segmented regions. In section 2.6, an architecture for bird classification using support vector machines is presented.

- *Actuation.* Actuation includes actions performed as a consequence of the preceding classification, which may or may not interact with the scene itself. This last point is not applicable, as the designed system is intended to be used offline.

The software prototype developed in the scope of this project was written in the Matlab$^{®}$ environment.

2.1 Preprocessing

As described above, the data used in this project consists of PNG images captured from the DVI output of a radar system deployed at the airport of Leipzig/Halle. The images have a resolution of 1280x1024 pixels and 24 bit color depth. An example of such an image is shown in figure 2.1. Besides the echoes and trails of detected objects, the images contain various other information that is part of the radar's PPI display (such as current settings, coordinates, position and orientation lines). In a first step, these display items are removed from the images using the distinct color ranges used by the individual display elements. Two sets of colors are defined: (1) those of display elements which do not interfere with the display portion containing the radar echoes (the background and those outside the radar "circle") and (2) those items that possibly collide with radar echoes, such as horizontal/vertical position lines. The former can be removed by setting pixels of the corresponding colors to zero while in the case of the latter, interpolation is necessary when radar echoes or trails are intersected. As only horizontal and vertical lines with a width of one pixel are encountered, interpolation is rather straightforward and is performed by replacing the conflicting pixels with the mean RGB values of the neighboring non-conflicting pixels. Interpolation is achieved by taking the maximum of two convolutions at the conflicting pixel positions, one of which takes the average between the pixels above and below, and one between the
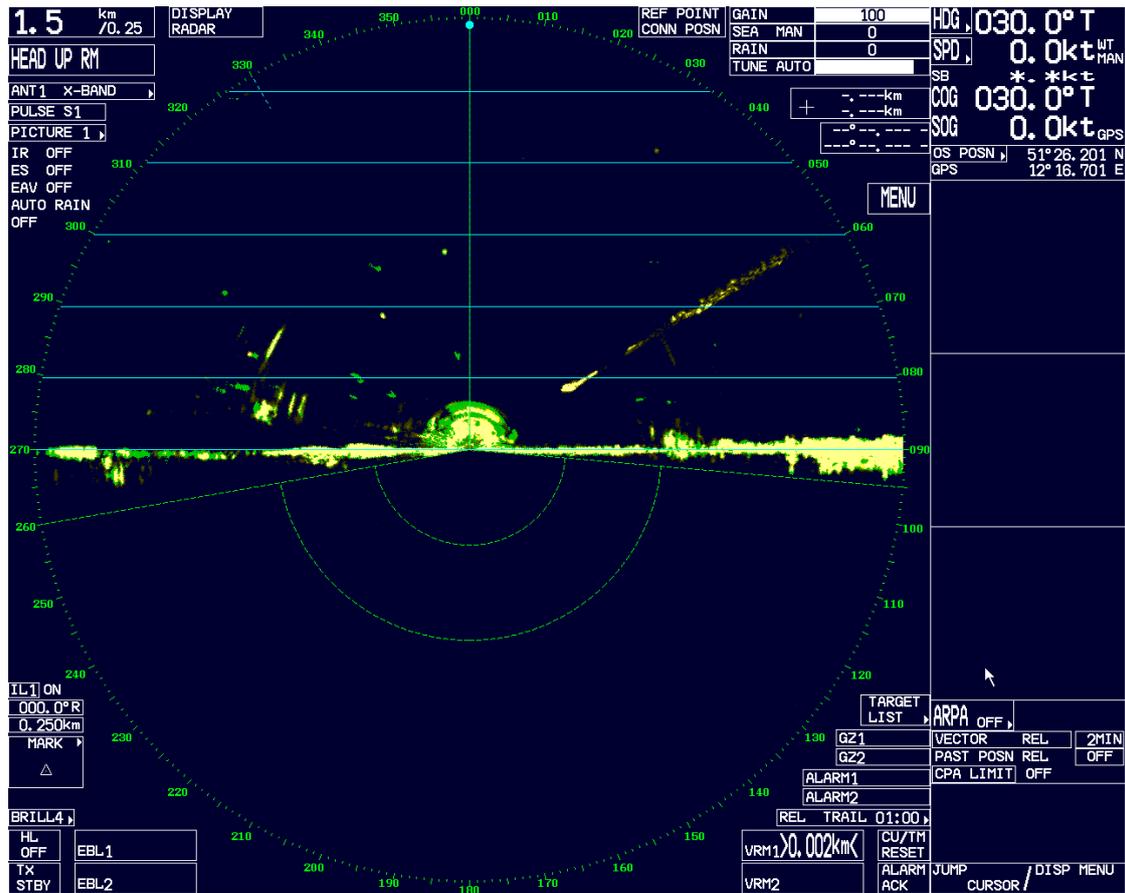
Fig. 2.1. Screenshot of the radar's PPI display, including display elements showing various information as well as horizontal/vertical position lines interfering with echoes.
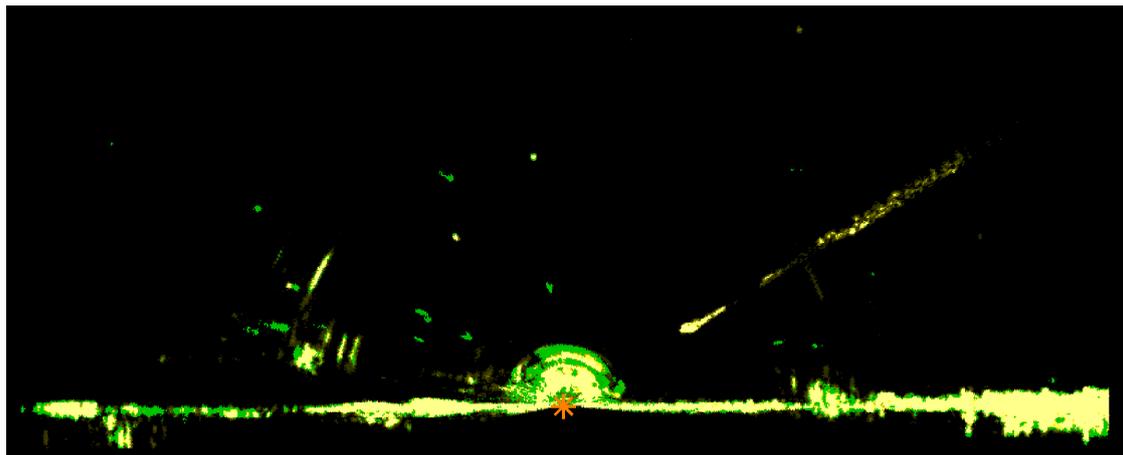

Fig. 2.2. Region of interest from the image shown in figure 2.1, after GUI removal. Current echoes are colored yellow, trails accumulated over the last minute are colored green. The radar position is shown as an orange asterisk.

pixels left and right of the center pixel. Consequently, the used convolution kernels
are

$$h_1 = \begin{pmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 0 \end{pmatrix} \quad h_2 = \begin{pmatrix} 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \end{pmatrix} \tag{2.1}$$

Figure 2.3 shows the result of interpolating in two image extracts containing echoes
intersected by PPI display elements.
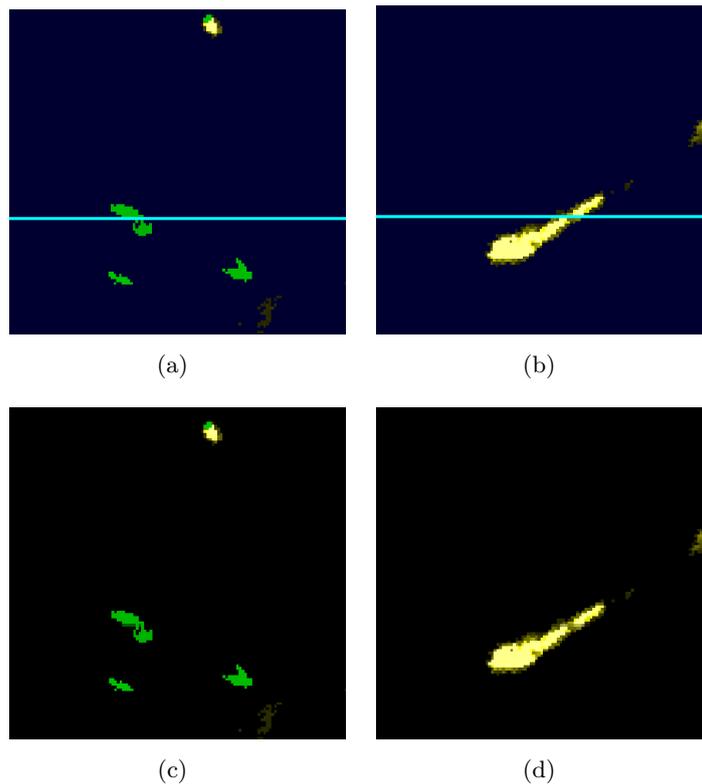

(a)   (b)

(c)   (d)

Fig. 2.3. Two examples of lines intersecting echoes (top row) and the same snippets after interpolation and GUI removal (bottom row).

As only a confined part of the captured images contains the radar signals with
the remainder of the PPI display showing menus and various information, a rectangular region of interest (ROI) is defined which all further processing is based
on.

The used radar images contain the current echoes as well as the trails of detected
objects, which are derived from accumulated echoes of previous scans. Targets and
trails use distinct RGB color ranges, allowing them to be separated resulting in
one image containing solely target pixels and one containing trail pixels.

2.2 Ground Clutter

The formation of ground clutter echoes poses a major problem for the detection and tracking of small objects in radar images. With vertical scanning, the affected area is usually confined to the very bottom of the image, as can be seen in figure 2.2. In order to deal with the effects of ground clutter, dynamic clutter maps are generated from the currently analyzed image sequence. As those image regions contaminated with ground clutter will be relatively stable over many images, it suffices to select a sample from an image sequence and inspect the distribution of non-zero pixels in this sample. After converting the RGB input images into grayscale images I_k[2], three clutter maps with different interpretations are generated.

For each pixel, the fraction of images in the sample set, in which this particular pixel is non-zero is determined

$$M_1(i,j) = \frac{1}{N_{sample}} \cdot \sum_{k=1}^{N_{sample}} B_k(i,j) \tag{2.2}$$

with N_{sample} being the size of the sample set and B_k being a binary image containing ones where the kth image of the sample is non-zero. Using a threshold $\tau_{affected} \in [0,1]$, the resulting image is binarized creating a mask of the region affected by ground clutter.

$$M_{affected}(i,j) = \begin{cases} 1 \text{ if } M_1(i,j) > \tau_{affected} \\ 0 \text{ otherwise} \end{cases} \tag{2.3}$$

This first map deliberately does not incorporate gray values, as clutter of low intensity should nevertheless be labeled as such.

Although the majority of ground clutter is rather stable, small variations at the clutter edges are common. Because bird echoes might be detected close to clutter edges, a second mask representing clutter core is created. As the center of clutter regions is generally of higher intensity than clutter borders, the mean intensity of clutter pixels is used to define a region than can be removed as definitive clutter before further processing. To this end, the ratio of each pixel's average gray value and the maximum gray value g_{max} of all images I_k in the sample is calculated.

$$M_2(i,j) = \frac{1}{g_{max}} \cdot \frac{1}{N_{sample}} \cdot \sum_{k=1}^{N_{sample}} I_k(i,j) \tag{2.4}$$

The clutter mask is then defined as follows:

$$M_{clutter}(i,j) = \begin{cases} 1 \text{ if } M_2(i,j) > \tau_{clutter} \wedge M_{affected}(i,j) \\ 0 \text{ otherwise} \end{cases} \tag{2.5}$$

where $\tau_{clutter}$ is a threshold in $[0,1]$. Both clutter masks ($M_{affected}$ and $M_{clutter}$) are dilated once and holes in the clutter masks are filled resulting in homogeneous masks, as shown in figure 2.4(b).

---

[2] Conversion from RGB to grayscale is performed using the Matlab function rgb2gray, which forms a weighted sum of the red, green and blue components with weights 0.2989, 0.5870 and 0.1140.
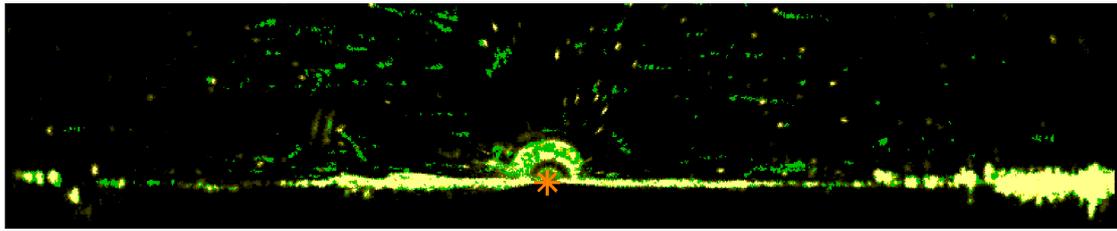
(a) Original image containing clutter (large, mostly connected region at the bottom and around the radar). Yellow pixels: Current targets. Green pixels: Trails accumulated by the radar system.
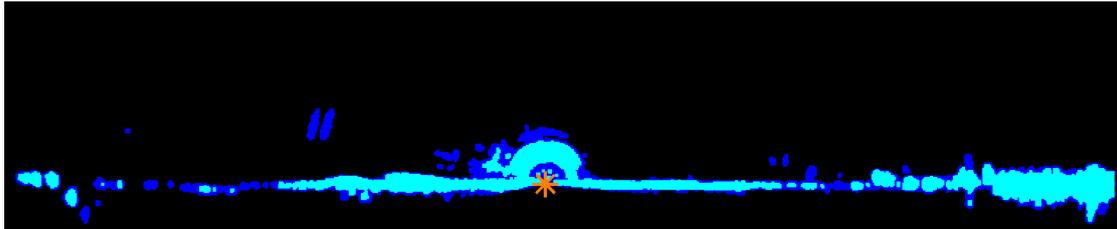
(b) Clutter masks of affected area (dark blue) and clutter core (light blue).

(c) Clutter intensity map and detected echoes. A strong purple overlay indicates high clutter intensity. Detected echoes are randomly colored.

Fig. 2.4. Clutter map generation.

Finally, a clutter intensity map $M_{intensity}$ attempts to supply a "fuzzy" description of the clutter area by applying average filters to the images of the sample set during and after gray value accumulation. $M_{intensity}$ is generated as follows:

1. Create an accumulator image of the same size as the images in the sample set and set all its pixels to zero
2. Iterate over images I_k in the sample set
   a) Find the maximum gray value of image I_k
   b) Normalize image I_k by dividing each pixel by the maximum gray value
   c) Create a mask by finding non-zero regions in I_k and dilate this mask once
   d) Filter the normalized image with a disk shaped averaging kernel where the mask is non-zero
   e) Add the filtered image to the accumulator image
3. Filter the accumulator image with a square kernel and normalize using the total number of images resulting in a map with elements between 0 and 1

Unlike the two clutter masks described before, the clutter intensity map does not give a clutter/no-clutter classification of pixels but a somewhat fuzzy ground clutter indicator (see figure 2.4(c)). If an object is detected in the vicinity of ground clutter, information from the clutter intensity map can be used as input to the classification system which can incorporate this information when making a decision regarding the object type.

2.3 Insect Contamination and Rain

Insect and precipitation echoes pose a major challenge to bird detection in radar images. Butterflies, dragonflies and moths are known to migrate in large numbers between northern Europe and tropical Africa and can cause contaminations of up to a few hundred meters (see figure 2.5(a)) [24]. Additionally, the vertical orientation of the radar results in an extremely strong sensitivity to precipitation, which can render images unusable, as shown in figure 2.5(b).
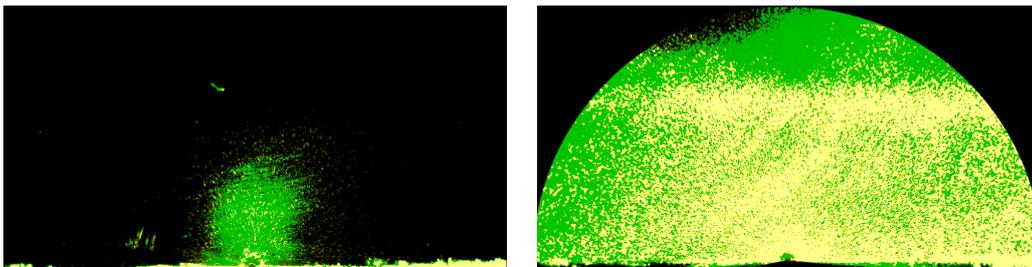

(a) Severe insect contamination up to ≈ 750m.   (b) Image completely filled with rain echoes.

Fig. 2.5. Examples of severe contamination caused by insects and rain.

Because contaminations caused by insects or rain manifest themselves in the form of numerous radar echoes cluttering the display, the contamination detection method uses only the current echoes converted to gray values, with trail information removed. The detection procedure is based on the assumption that (1) insect contaminations are limited to an area relatively close to the radar, (2) that if a contamination is present, the area it affects will be a large contiguous region and (3) that the number of echoes in this region will be high and scattered.

The first step of the contamination detection procedure consists of finding non-zero regions in the input image followed by ten-fold dilation of the resulting binary image, with the intention of creating a large connected region forming a mask of the area affected by insects. After filling potential holes, 4-connected components are labeled, resulting in an image similar to the one in figure 2.6(b), and their centroids and area are determined. The largest region detected is used as a mask to select the potentially contaminated region from the original gray image. An insect intensity map is produced by filtering the selected region with a relatively large disk-shaped averaging filter (see figure 2.6(c)).
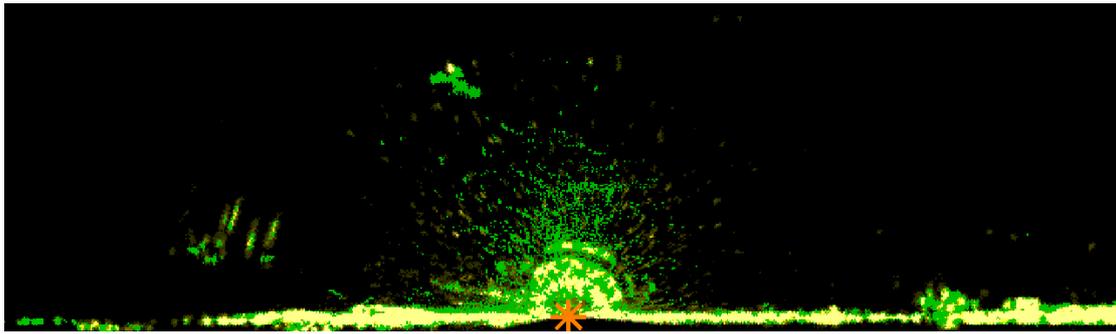
(a) Original image contaminated by insects.
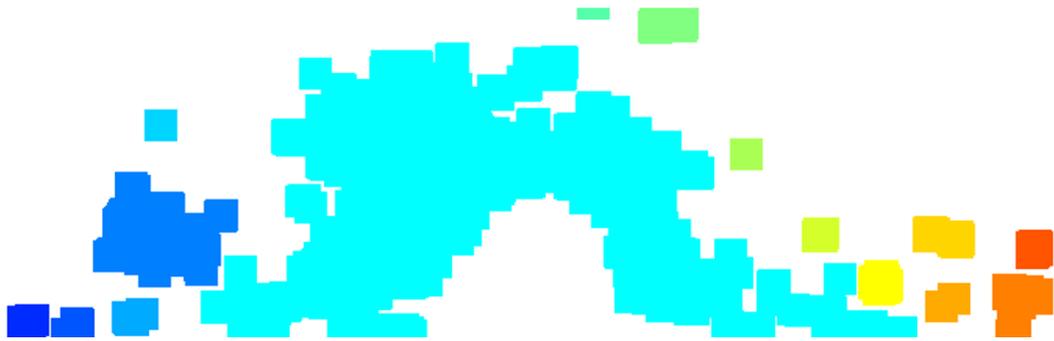
(b) Potentially contaminated regions (randomly colored). The largest region is selected and, if it fulfills certain criteria, is marked as contaminated.

(c) Insect intensity map. The region which seems to be contaminated is highlighted. A strong blue overlay indicates a high contamination level.

Fig. 2.6. Insect contamination detection.

In the next step, it is decided, whether the image should be flagged as contaminated based on the region segmented above. If the euclidean distance between the centroid of the potentially contaminated region and the radar position is larger than a threshold (and the region is not exceedingly large) or if the region is too small, the contamination hypothesis is rejected and the image is marked as "not contaminated". If this is not the case, the distribution of echoes in the potentially contaminated region is analyzed.

As insects cause significant amounts of low intensity echoes, all contiguous non-zero regions containing pixels with an intensity greater than 98% of the maximum possible value are removed from the original gray value image before the next steps. In order to assess the distribution of echoes in the region in question, connected components in this region are detected and the euclidean distance of each echo to its nearest neighbor is computed. The spatial variance of the echoes is quantified using the aggregation index presented in [33], which is based on the relationship of the observed average distance of each object to its nearest neighbor and the expected average distance if the echoes are randomly distributed.

$$R = \frac{\overline{r}_{observed}}{\overline{r}_{expected}} \tag{2.6}$$

The expected average distance to the nearest neighbor in case of a random distribution of N_e echoes over an area A is

$$\overline{r}_{expected} = \frac{1}{2 \cdot \sqrt{\frac{N_e}{A}}} \tag{2.7}$$

If the aggregation index for the potentially contaminated region and the maximum intensity of the previously calculated insect intensity map are each above a threshold, it is assumed, that the inspected image is contaminated.

## 2.4 Target Detection and Target Properties

2.4.1 Target Segmentation

Having removed regions contaminated with ground clutter, object detection is performed. As a first step, non-zero regions in the image containing the radar targets $I_{targets}$ and the image containing the trails I_{trails} are extracted, resulting in binary images $B_{targets}$ and B_{trails}. In some cases, trail pixels might interfere with target echoes, resulting in holes or distorted echoes (see figure 2.7(a)). Therefore, an interpolation measure is taken prior to segmenting the echoes in $I_{targets}$. First, pixels which might need interpolation are identified. Unset pixels having 4 or more neighboring non-zero pixels are determined:

$$B'_{interp}(i,j) = \begin{cases} 1 \text{ if } \neg B_{targets}(i,j) \wedge (\sum_{b \in N_8(i,j)} B_{targets}(b)) \geq 4 \\ 0 \text{ otherwise} \end{cases} \tag{2.8}$$

where $N_8(i,j)$ is the 8-neighborhood of pixel $B_{targets}(i,j)$. This operation is repeated using $B'_{interp} \vee B_{targets}$ instead of $B_{targets}$ resulting in mask B_{interp}. The targets image is then interpolated as follows:

$$I'_{targets}(i,j) = \begin{cases} \frac{1}{8} \cdot \sum_{g \in N_8(i,j)} I'_{targets}(g) \text{ if } B_{interp}(g) \wedge B_{trails}(g) \\ I_{targets}(i,j) \qquad\qquad\qquad \text{otherwise} \end{cases} \tag{2.9}$$
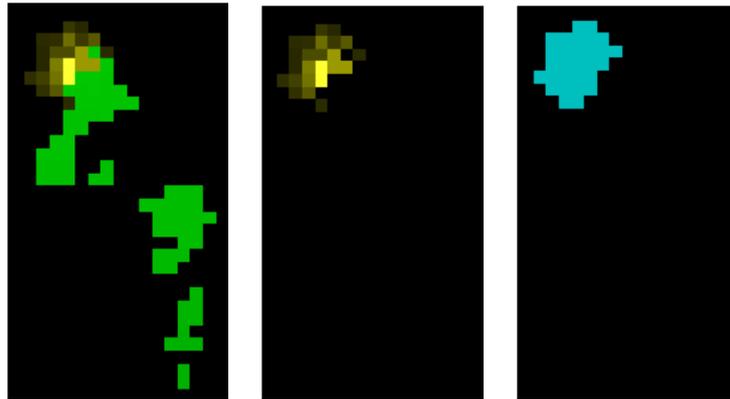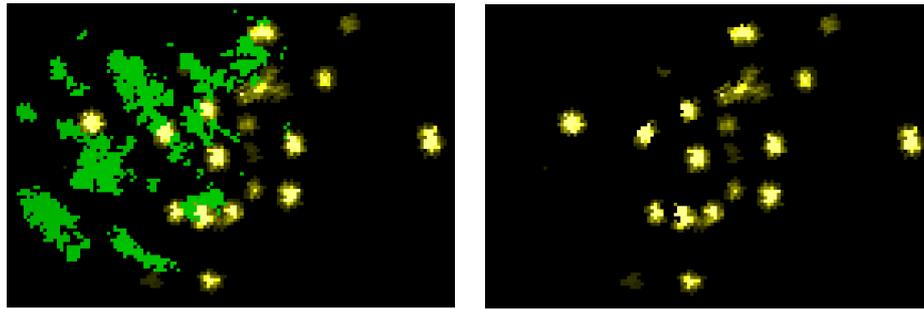
(a) Snippet of bird echo with trail. (b) Echo without trail. (c) Segmented echo after interpolation.

**Fig. 2.7.** Target echo interpolation.

Next, potential insect contamination is detected using the method described in section 2.3, resulting in an insect contamination map $M_{insects}$ and a flag indicating whether there is a contamination or not. If there is a contamination, those regions of $I'_{targets}$ and I_{trails} for which the contamination intensity map exceeds a threshold are excluded from further processing as it is assumed, that bird detection in these regions is futile.

4-connected components are now detected in the filtered and clutter-free targets image, resulting in a number of segmented radar echoes. A common phenomenon is that two (or more) echoes are merged into one because of the limited resolution of the radar, resulting in the detection of one large object. This larger object often consists of regions of high intensity separated by a belt (or multiple belts) of low intensity pixels (see figure 2.8(b)). As the images used in this project were captured at one minute intervals, it is not possible to discriminate multiple objects merged into one echo based on previous or future images. It is therefore assumed that, if, after setting pixels below 0.8 times the mean intensity of the echo to zero, two or more contiguous regions above a certain size remain, the detected object must originate from multiple targets and it is thus split into these smaller objects (see figure 2.8(c)). If thresholding the echo does not result in more than one object, the original object is kept. An exception is made, if the centroid of the thresholded object differs significantly from the centroid of the original echo. In this case, it is assumed that the actual target corresponds to the thresholded echo and the low-intensity portion of the echo is dismissed. Very small echoes a removed per se.

The targets having been segmented, various properties of these targets can be computed.

(a) Snippet of bird flock echoes with trails.

(b) Echoes without trails. A merged echo can be seen in the lower center of the snippet.
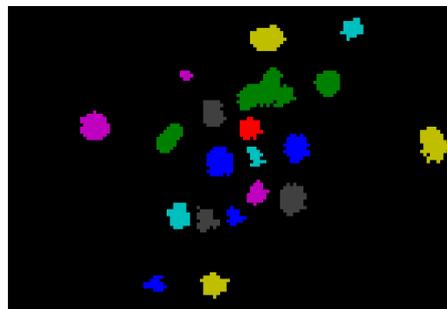
(c) Segmented targets after splitting. Bird echoes are randomly colored.

Fig. 2.8. Splitting of echoes.

2.4.2 Trail Detection

As described above, this project is based on images captured once per minute from the radar system's PPI display. While this keeps memory requirements low, it has several drawbacks. One of these drawbacks is that if tracking of targets is required, this can only be achieved by analyzing the trail information accumulated and plotted by the radar device. Additionally, birds are uncooperative targets that often turn sharply, soar in tight circles and take-off and land again after flying short distances ([34]), making this trail information ambiguous and often difficult to interpret. Even if the benefits of tracking targets in radar images obtained from vertically operating radar is debatable (see section 1.2), some hints regarding flight paths can nevertheless be derived and a strong, distinct trail is a good indicator that the corresponding echo might be a bird.

Another motivation for detecting the flight paths of targets comes with the fact, that the great velocity of planes taking off or landing can cause a number of echoes to be detected along the flight path, which may mistakingly be classified as birds (see figure 2.9). If plane trajectories can be identified, the echoes found along these trajectories can be marked as such, preventing a misclassification.
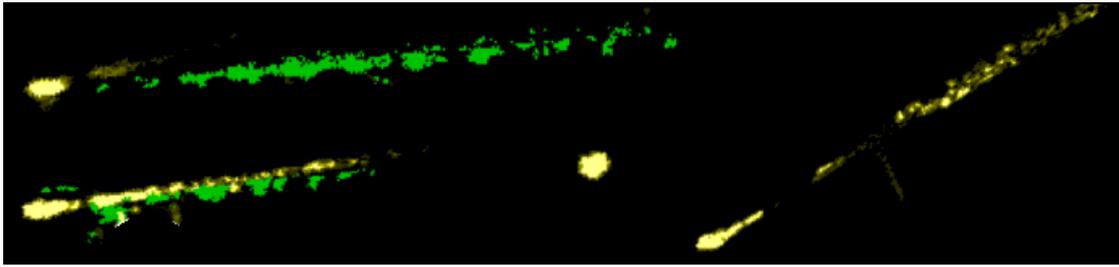
Fig. 2.9. Four examples of plane echoes of varying appearance (from different source images).

In the following, a modified Hough transform which roughly determines the flight path of targets is presented.

The Standard Hough Transform

The Hough transform is a mapping of pixel coordinates (i, j) to coordinates in a parameter space (termed "Hough space"), each of which represent a straight line passing through the original image pixel. The transform is based on the assumption, that any non-zero pixel in a binary image can hypothetically be part of an infinite set of straight lines passing through it at arbitrary angles. Based on the normal form $r = x \cdot \cos \omega + y \cdot \sin \omega$ of the line equation, any line in the image can be described using the Hough space coordinates (r, ω), where r is the perpendicular distance from the origin to the line and ω is the angle of that perpendicular line to the x axis [35].

When detecting lines, a degree of quantization of the hough space coordinates (r, ω) is decided upon and a Hough image H is created. For each non-zero pixel in the binary image B, all quantization steps of ω are inserted into the line equation $r = x \cdot \cos \omega + y \cdot \sin \omega$ together with the current pixel's coordinates, thus obtaining the corresponding values of r. In each iteration, $H(r, \omega)$ is incremented. When all pixels have been processed, peaks in H indicate large numbers of collinear pixels in B, suggesting the existence of lines (or line segments) with parameters (r, ω) [35].

Modified Hough Transform

In the case of trail detection, be it identification of bird trails or plane trajectories, only lines intersecting the target echo are of interest. Therefore a ROI with a radius of N_{ROI} pixels is defined around the examined object, imposing a maximum trail length. It would be possible to use the centroid of the examined object as origin in a standard Hough transform and then select Hough space peaks with low r (corresponding to lines passing close by the object centroid) as trails. However, because the orientation of vectors connecting the echo and non-zero pixels in the ROI can be easily determined and because it seems reasonable to apply a weighting that emphasizes trail pixels close to the echo, a customized transform is used.

A resolution is defined for all possible trail angles θ, resulting in an accumulation vector h_θ with N_θ bins. A second vector h_{count} of the same length keeps track of the number of trail pixels that contributed to each bin of h_θ. Additionally, a $N_\theta \times N_{ROI}$ matrix H_{dist} stores for each bin in h_θ, how much the pixels of a given distance contributed to that particular bin.

For every trail pixel in the ROI, the angle to the centroid of the examined echo is determined and the corresponding bin in h_θ is identified. As a smoothing measure, neighboring bins of the accumulators h_θ, h_{count} and H_{dist} are simultaneously incremented. In the case of h_θ, the value added to each bin depends on the distance of the current pixel to the echo (d) and on a one-dimensional symmetric Gaussian filter f_σ. Given the filter kernel radius ρ, standard deviation σ and $k = 1 \ldots (2\rho + 1)$, the filter is constructed as follows

$$g_\sigma(k) = e^{\frac{-(k-\rho-1)^2}{2\sigma^2}} \tag{2.10}$$

$$f_\sigma(k) = 1 + \frac{g_\sigma(k)}{\sum_{l=1}^{2\rho+1} g_\sigma(l)} \tag{2.11}$$

i being the bin corresponding to the current angle, h_θ is updated as follows

$$h_\theta(j) = \begin{cases} h_\theta(j) + f_\sigma(j-i) \cdot (1 - \frac{d}{N_{ROI}})^\eta & \text{if } |j-i| \le \rho \\ h_\theta(j) & \text{otherwise} \end{cases} \tag{2.12}$$

where $j = 1 \ldots N_\theta$ and η is a distance weight (e.g. $\eta = 4$) defining how much influence distance should have on the accumulation process. Note that in the special case where θ is close to 0° (360°), the bins at the other end of the vector are incremented accordingly. Similar to h_θ, the counting vector h_{count} is incremented:

$$h_{count}(j) = \begin{cases} h_\theta(j) + 1 & \text{if } |j-i| \le \rho \\ h_\theta(j) & \text{otherwise} \end{cases} \tag{2.13}$$

Given the rounded distance $d^* = round(d)$ from the object centroid to the current pixel, H_{dist} is updated as follows:

$$H_{dist}(j, d^*) = \begin{cases} H_{dist}(j, d^*) + f_\sigma(j-i) & \text{if } |j-i| \le \rho \\ H_{dist}(j, d^*) & \text{otherwise} \end{cases} \tag{2.14}$$

After accumulation, the most likely trail angle is derived from the index i_{max} of the maximum element of h_θ. Additionally, a normalized trail score is calculated by dividing $h_\theta(i_{max})$ by the value it would reach, if all pixels in the ROI were non-zero.

Finally, the weighted mean distance of pixels contributing to the winner bin is calculated as follows:

$$\overline{d} = \frac{\sum\limits_{d' | H_{dist}(i_{max}, d') \ne 0} H_{dist}(i_{max}, d') \cdot d'}{\sum\limits_{d' | H_{dist}(i_{max}, d') \ne 0} d'} \tag{2.15}$$
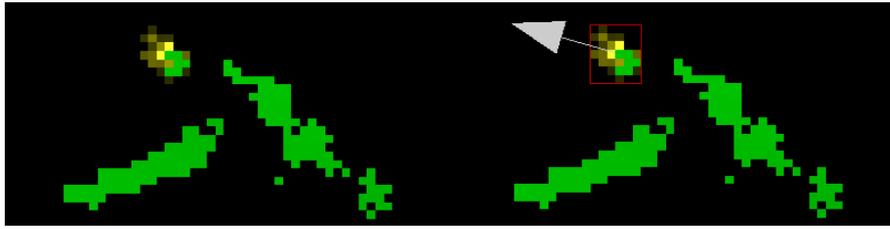
Fig. 2.10. Bird echo with trail. Left: Original echo. Right: Detected trail. The arrow highlights the estimated flight direction. ROI radius from the target centroid is 50px.
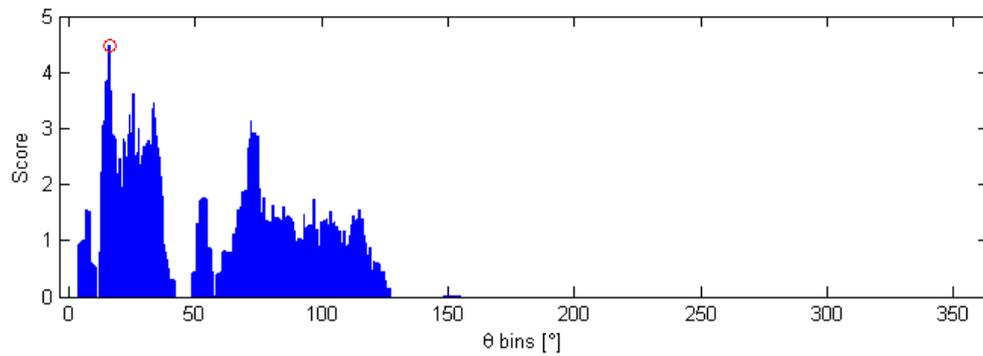
(a) Plot of h_θ. The maximum value is highlighted. The peaks between 0° and 50° correspond to the actual trail while those between 50° and 125° correspond to the second trail visible.
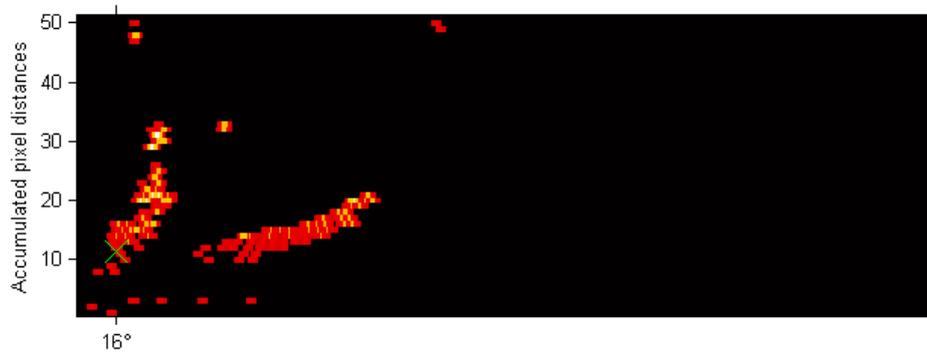
(b) Map H_{dist} showing for each angle how many pixels from what distance contributed to the angle's bin. Bright regions indicate many contributing pixels. The × shows the weighted mean distance $\bar{d} \approx 11$ at the winner angle of 16°.

Fig. 2.11. Accumulation vector and distance map for the echo shown in figure 2.10.

Under certain conditions, the trail corresponding to the winner bin is dismissed. This is the case

- if the normalized trail score is extremely low,
- if the normalized trail score is below a threshold and the weighted mean distance $\overline{d}$ of the trail is above a threshold, indicating that the trail belongs to a different echo or
- if the the sum of h_{count} for the winning bin and its left and right neighbor bin is less than a threshold (relative to the size of the ROI), i.e. hardly any pixels contributed to the winning trail.

A problem occurs, when apart from the examined echo, the ROI contains several other echoes whose trails might be mistaken for the trail of the examined echo. This will result in peaks in h_θ at the angles in question, even if the trails are in the shadow of the surrounding echoes. Figure 2.12(a) shows such a constellation. One

(a) Several bird echoes. A trail for the central echo (red) is sought.

(b) Surrounding echoes.

(c) "Line of sight" of the central echo in all directions (red indicates blocked sight).
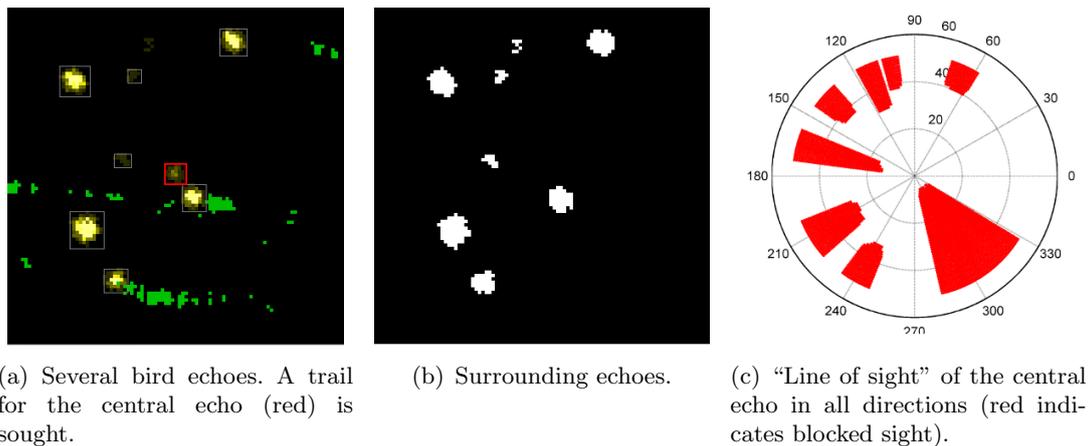
Fig. 2.12. Dealing with trails of other targets in the vicinity of a bird echo.

solution which has been considered consists of adding a preprocessing step to the trail detection, in which for each bin in h_θ, the "line of sight" at the corresponding angle is determined. This is done by iterating over all target pixels in the ROI, calculating the angle and distance to the examined echo and updating the minimum distance at that angle in case a new minimum is found. When performing the actual trail detection, trail pixels that are in the shadow of other targets (i.e. the line of sight value at the given angle is less than the distance of the trail pixel to the examined target) are not counted. Another, more sophisticated approach for dealing with this problem would be to compare the results of the trail detections of neighboring echoes. If a trail has been assigned to more than one echo, the decisions leading to the conflict could be revisited. As a result, the trail in question could be assigned to the echo with the highest trail score or, if indicated,

one could conclude that the decisions were in fact correct and the trail belongs to multiple birds flying in formation. Furthermore, this approach has the advantage, that it is equally applicable when multiple echoes are detected close together, but flying in different directions, such as is the case with three echoes in the bottom part of figure 2.13.
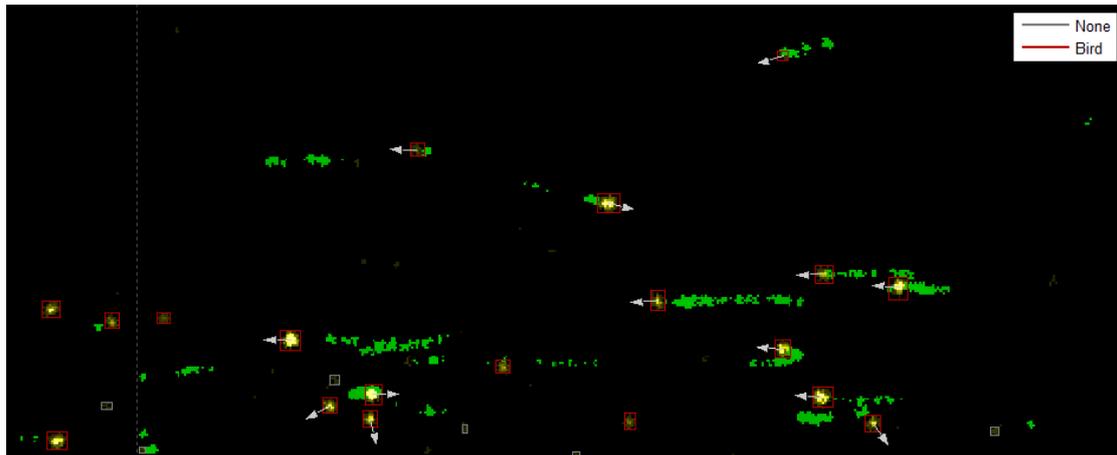

Fig. 2.13. Trails detected for bird echoes. Arrows show the estimated flight direction. Note that some trails are not detected correctly, indicating there is room for improvement.

Figure 2.14 shows a radar image containing plane echoes with the plane's estimated trajectory superimposed. When performing trail detection to identify plane trajectories, using both trail pixels and target pixels is necessary, as planes can produce significant amounts of both pixel types (as can be seen in the examples in figure 2.9). Additionally, in the case of plane trajectory identification, skeletizing the binary trail image prior to running the above detection method proved useful while detection of bird trails performs better without skeletization.

The trail detection procedure was implemented in a C++ MEX-file resulting in a computation time eight times faster compared to pure Matlab code.

2.4.3 Object Properties

Application-specific Properties

After target segmentation and trail detection, a set of properties is calculated for each detected echo. Several application-specific features introducing some level of semantic interpretation of the inspected object and the surrounding area were designed.

- *Distance to radar and object location.* The euclidean distance of the centroid of the object from the radar position is determined, as the interpretation of echoes can vary with distance. For example, a weak target echo detected at a location far away from the radar is most likely a bird, as insects are only detected up
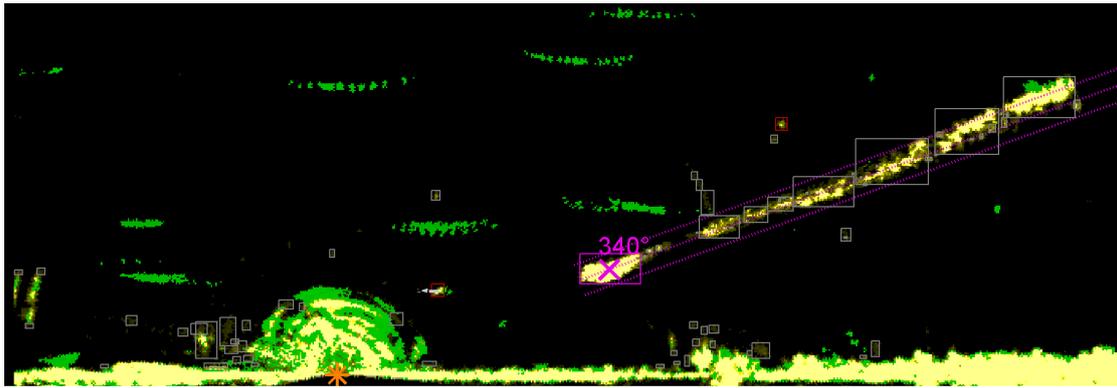
(a) Radar image extract showing plane echoes and the determined trajectory.

(b) Skeletized image extract used as input for the trail detection.

Fig. 2.14. Plane trajectory estimation.

to a certain distance. For objects which are closer, classification might be less straightforward with insect contamination being common in the vicinity of the radar.

The altitude of the detected object is of great importance for ornithological assessment of the data and can potentially be used as input to a classifier, as insects will usually be contained to lower height bands.

The position of the target in respect to the main direction of the radar is also determined. It is expressed as an angle to the left or the right of the vertical line up from the radar location.

- *Trail orientation and trail level.* The results of the trail detection as described in section 2.4.2 are used as object properties in form of (1) the trail orientation, which is the angle at which the most likely trail was found and (2) the trail level, which is the accumulation score that caused this angle to be selected. The trail orientation is set to -1.0, if no trail could be detected.

- *Clutter level.* The clutter level is a measure describing the amount of ground clutter at the position where the object was detected. It is defined as the average clutter intensity of the pixels inside the bounding box of the object. Intensities

are looked up in the current clutter intensity map (see section 2.2 for details on clutter map generation).

- *Contamination level.* The contamination level is similar to the clutter level, measuring contamination by insects or precipitation instead of ground clutter. It is defined as the mean contamination intensity of the object's pixels, which is looked up in the insect intensity map (see section 2.3 for details on insect intensity map generation).

- *Aggregation index.* The concept of the aggregation index has already been presented in section 2.3: It measures the relationship of the observed average nearest neighbor distances between objects and the expected average distance if the objects are randomly distributed. Here, it is designed to assess how trail pixels are distributed in a ROI around an echo. If the aggregation index is high, trail pixels are scattered over the ROI, indicating a contamination with insects or the presence of trails belonging to other targets. Conversely, if the aggregation index is low, trail pixels are clustered in a certain region of the ROI. As the calculation of nearest neighbor distances comes with high computational costs, it is only feasible for relatively small pixel numbers. However, because when a certain proportion of the ROI is filled with trail pixels, a contamination can be assumed and calculating the aggregation index is deemed unnecessary. In this case, it is set to a high value.

- *Trail pixel density.* The trail pixel density is defined as the ratio of trail pixels in a ROI around the object and the total area of this ROI. High densities can be an indicator of contamination or of increased activity of other targets in the vicinity of the echo in focus.

Morphological Properties

Besides the application-specific features described above, the developed software prototype is capable of computing various morphological measures. For the following sections A shall denote the area of an object, P its perimeter, N_{pixels} the number of object pixels, and L_{min} and L_{maj} the length of the minor (major) axis of the ellipse that has the same normalized second moments as the object [36]. Based on these standard properties available in Matlab, various other morphological measures can be defined.

- Maximum Feret's diameter
  The maximum Feret's diameter (or maximum caliper) measures the maximum euclidean distance between any two points of the object [37].

$$FeretsDiameter = max(\|p_i - p_j\|) \tag{2.16}$$

  where p_i and p_j are object pixels and $i, j = 1 \ldots N_{pixels}$. Note that the maximum Feret's diameter is strongly correlated with L_{maj} (which is not surprising).
- Bounding Box Area
  The bounding box area is defined as the area of the smallest rectangle (not tilted) containing all object pixels.

- Roundness measures
  Both compactness and roundness are one only for a circle and less for any other shape, as they both exploit the fact, that a circle is the most compact shape there is [36, 37].

$$Compactness = \frac{4\pi \cdot A}{P^2 + 1} \tag{2.17}$$

$$Roundness = \frac{4 \cdot A}{\pi \cdot FeretsDiameter^2} \tag{2.18}$$

- Axis ratio
  The axis ratio of an object is defined as the ratio of the length of the major and minor axes of the ellipse having the same normalized second moments as the object, and is therefore a measure of longness.

$$AxisRatio = \frac{L_{maj}}{L_{min}} \tag{2.19}$$

- Convexity and concavity
  Convexity is defined as the perimeter of the object's convex hull[3] divided by the perimeter of the object itself. Concavity is defined as the difference of the area of the convex hull and the area of the object [37].

$$Convexity = \frac{P_{ConvexHull}}{P} \tag{2.20}$$

$$Concavity = A_{ConvexHull} - A \tag{2.21}$$

- RFactor
  The R factor is defined as the ratio of the perimeter of the object's convex hull and the maximum Feret's diameter [37].

$$RFactor = \frac{P_{ConvexHull}}{\pi \cdot FeretsDiameter} \tag{2.22}$$

Additional measures considered were median intensity of an echo and intensity variance.

2.5 Feature Distributions and Feature Selection

Having computed a set of object properties for each potential bird echo, the question is which properties are suited best for class separation. In order to assess the distributions of feature values, several visualization and dimensionality reduction techniques were added to the software prototype. As a basic method of inspecting feature distributions, scatter plots of a subset of features can be created in the software. Additionally, principal component analysis can be performed and self-organizing maps can be used to find correlations and clusters in the data. The principles of these two techniques are summarized in the following section.

---

[3] The convex hull of a region is the smallest convex polygon that can contain the region [36]

2.5.1 Principal Component Analysis

Principal component analysis (PCA) is probably the most popular multivariate statistical technique and is used by almost all scientific disciplines. The central idea behind it is to reduce the dimensionality of a data set in which there are a large number of interrelated variables, while retaining as much as possible of the variation present in the data set. This reduction is achieved by transforming the data to a new set of variables, the principal components (PC), which are uncorrelated, and which are ordered so that the first few retain most of the variation present in all of the original variables [38, 39].

If x is a vector of p random variables, the first step to finding the principal components of x is to look for a linear function $\alpha_1^t x$ of the elements of x having maximum variance. α_1 being a vector of p constants $\alpha_{11}, \alpha_{12}, \ldots, \alpha_{1p}$, we get

$$\alpha_1^t x = \sum_{j=1}^{p} \alpha_{1j} \cdot x_j \tag{2.23}$$

Next, look for a linear function $\alpha_2^t x$, uncorrelated with $\alpha_1^t x$ (i.e. orthogonal to $\alpha_1^t x$) having maximum variance, and so on, so that at the kth stage a linear function $\alpha_k^t x$ is found that has maximum variance subject to being uncorrelated with $\alpha_1^t x, \alpha_2^t x, \ldots \alpha_{k-1}^t x$. The kth derived variable $\alpha_k^t x$ is the kth principal component. Up to p PCs could be found, but it is hoped, in general, that most of the variation in x will be accounted for by $m < p$ PCs. The values of the resulting principal components are called factor scores and can be geometrically interpreted as the projections of the observations onto the principal components [38, 39].

If successful, i.e. the first few PCs account for most of the variation in the data set, those input variables having low factor scores in the first principal components can be eliminated.

2.5.2 Self-Organizing Maps

Self-Organizing Maps (SOMs) are artificial neuronal networks based on unsupervised learning. A SOM consists of neurons representing weight vectors (prototype vectors, codebook vectors) organized on a regular low-dimensional grid. A SOM can be thought of as a net which is adjusted to the properties of the data cloud (the training data). In the trained map, neighboring neurons receive similar weight vectors that may form clusters which correspond to clusters in the input data. The training procedure roughly works as follows: The d-dimensional weight vectors $w_i = \left(w_i^1 w_i^2 \ldots w_i^d\right)^t$ of the map's neurons are initialized with random values. During the training process, input vectors $x_j = \left(x_j^1 x_j^2 \ldots x_j^d\right)^t$ are randomly drawn from the training data set and the neuron whose weight vector is closest to the input vector, the best-matching unit (BMU), is determined:

$$\|x_j - w_{BMU}\| = \min_i(\|x_j - w_i\|) \tag{2.24}$$

Once the best-matching unit has been identified, the underlying weight vector w_{BMU} as well as the weight vectors of the neighboring neurons are drawn closer to the input vector x_j, according to a given learning rate and neighborhood kernel. Iteratively presenting training vectors to the SOM causes the map to gradually adjust to the training data [40]. After training the SOM with selected input features,

(a) U-matrix and component planes.

(b) Hit/color plane with two clusters.
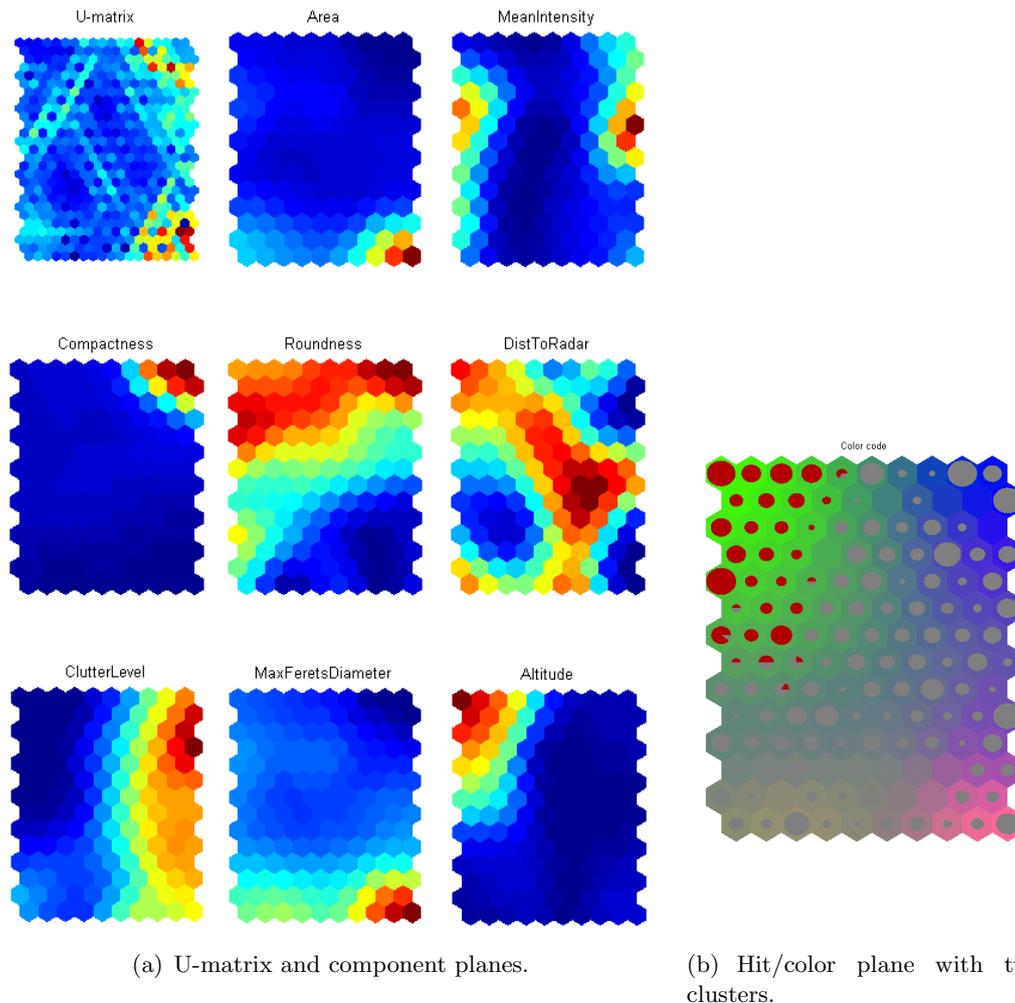
Fig. 2.15. SOM example for a simple problem. The map is trained with echoes classified as clutter (gray) and non-clutter (red).

the map is visualized with different information superimposed. (1) The *U-matrix* (figure 2.15(a)) visualizes distances between neighboring map units. High values in the U-matrix indicate a cluster border whereas low values are characteristic for the clusters themselves. (2) *Hit planes* (figure 2.15(b)) show the distribution of best-matching units (the neuron whose weight vector is closest to an input vector) when presenting training or testing data to the trained map. The hits are shown as pie charts drawn on top of the neurons with the manual classification shown

as pie colors and the size of the pie chart indicating how many data samples hit a certain neuron. In the example in figure 2.15, a cluster has formed in the top left corner of the map. (3) *Component planes* (figure 2.15(a)) color code the values of each neuron's weight vector for a single feature. When looking at clusters on hit planes, component planes show which feature values are characteristic for a certain cluster and which features are redundant or irrelevant. In the example, one could derive that *Area* (top center component) and *MaxFeretsDiameter* (bottom center component) are correlated. One can also see, that altitude and clutter level separate the two classes pretty well. (4) *Color planes* (figure 2.15(b)) show the SOM in colors that represent the similarity between neighboring neurons according to a certain distance measure. Uniformly colored areas indicate that the underlying prototype vectors are similar. In combination with a hit plane, one can see whether the formed hit clusters are well-separated. The top left corner of the example shown in figure 2.15(b) is an example of such a uniform cluster.

2.6 Classification

After radar echoes have been segmented and ground clutter as well as contaminated regions have been labeled, classification can be performed. Based on the above observations, various classification scenarios need to be taken into consideration (see figure 2.16 for examples).

- *Detecting birds in clear weather without contamination.* This is the default scenario which consists of identifying single birds in regions without interferences caused by precipitation, insects or ground clutter (see figure 2.16(a)). Most echoes observed in such a situation will be birds (if above a certain size).
- *Detecting birds in areas contaminated by insects or light precipitation.* In most cases when part of the observed volume is contaminated, it is still possible to detect birds, especially in boundary regions with a lower contamination level (see figure 2.16(b)). High echo intensity, typical morphology (round shape) and distinctive trails are hints suggesting that the target might be a bird.
- *Detecting birds in the vicinity of ground clutter.* Birds flying particularly low, landing/sitting on objects in the ground clutter or birds crossing miscellaneous clutter regions form another challenge to the classification system. It is often difficult to detect such targets in single images, when the bird echo merges with clutter, as shown in figure 2.16(c). In such a case, detection can be ameliorated by including temporal information, which with images captured once per minute is supplied in form of trails accumulated by the radar system. Ideally, this information would be obtained by tracking targets in subsequent images captured at a higher rate, thus enabling the software to observe echoes of birds landing and taking off or crossing clutter above ground (see figure 2.16(d)).
- *Classification of airplanes and identification of plane trail echoes.* Compared to the number of bird echoes found in the radar images, planes are a rare sight ([28] estimates that planes are encountered in $0.5 - 1.0\%$ of the images). Plane echoes can in most of the cases be easily distinguished based on a significantly

(a) Birds without interferences

(b) Bird flying through insect contamination

(c) Bird echo merged with ground clutter

(d) Birds flying through clutter

(e) Plane producing trail echoes

(f) Bird flock (compact appearance)
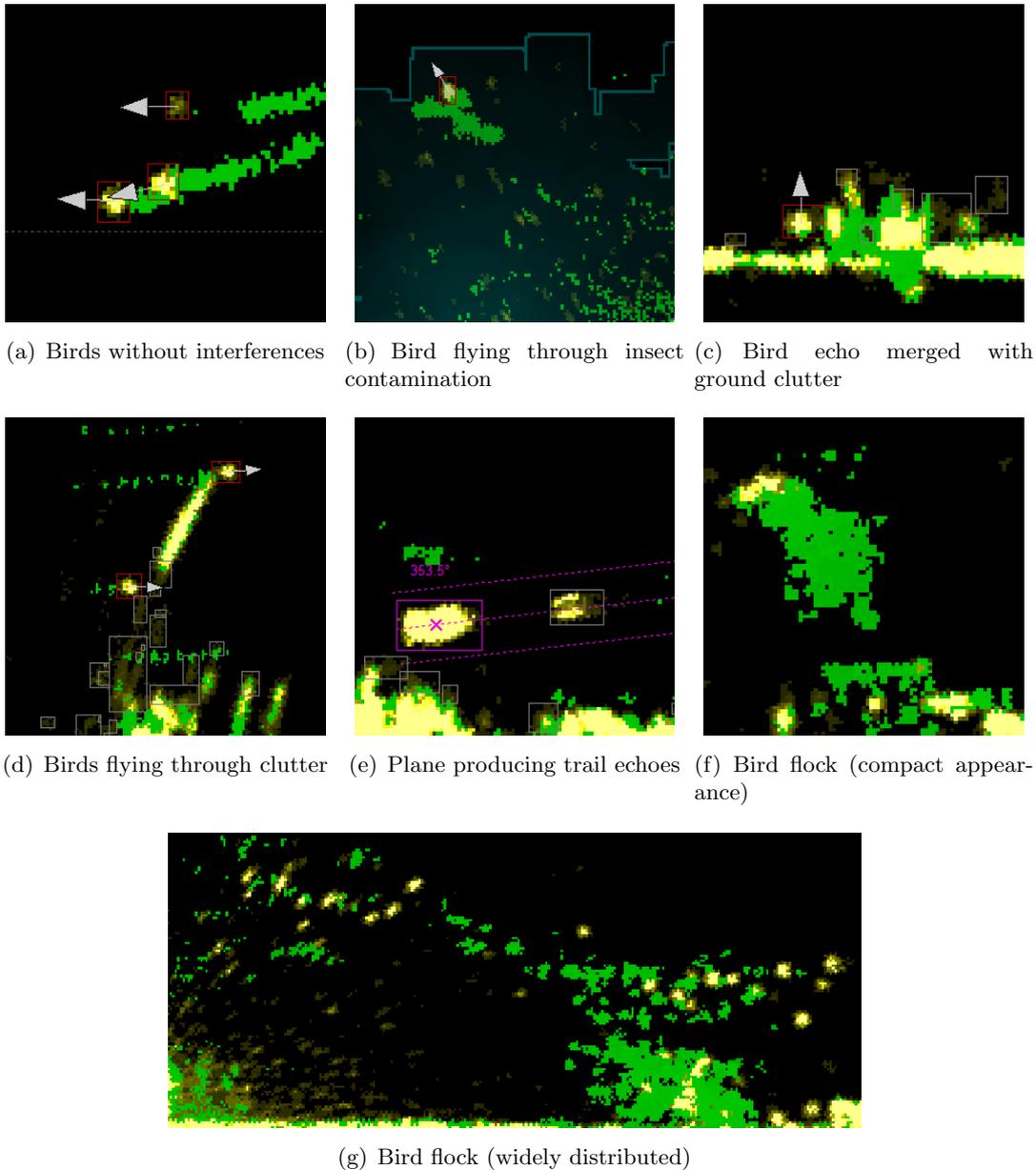

(g) Bird flock (widely distributed)

Fig. 2.16. Examples of various situations encountered in radar images.

larger size and higher intensity compared to other objects. However, because of the great flying speed, planes can potentially produce numerous trail echoes on their trajectory, which might be mistaken for birds (see figure 2.16(e) or 2.9). Therefore, the flight trajectory must be determined in order to label the plane trail echoes as such. If images captured at higher rates are available, planes can be tracked in subsequent images thus improving trajectory estimation. With the images used in this project, trajectory estimation can be derived only from target echoes and trails accumulated by the radar (see section 2.4.2).

- *Classification of bird flocks.* Birds can fly in flocks which may exceed several thousand individuals (e.g. [10]), producing echoes that are very different from those of single birds. Because of the variability in flock size and aspect (see figure 2.16(f) and 2.16(g)), it is difficult to characterize flock echoes when working with single images. Again, if images captured at a higher frequency were available, objects could be tracked in subsequent images and thus distinguishing flocks from airplanes, clutter and insects could be simplified.

Considering the numerous scenarios and echo appearances encountered, using a multi-level classification system seems appropriate as each system level can be adapted to its individual purpose. This means treating echoes from contaminated or cluttered areas differently from "clean" echoes thus avoiding that too much is demanded of a single level classifier. Figure 2.17 outlines how such a multilevel classifier could be designed. As the flowchart shows, the idea is to detect planes and bird flocks first, then performing bird detection on the remaining echoes. Depending on the spatial context of the echo (clutter, contamination or clean), different classifiers are chosen to make the final decision.
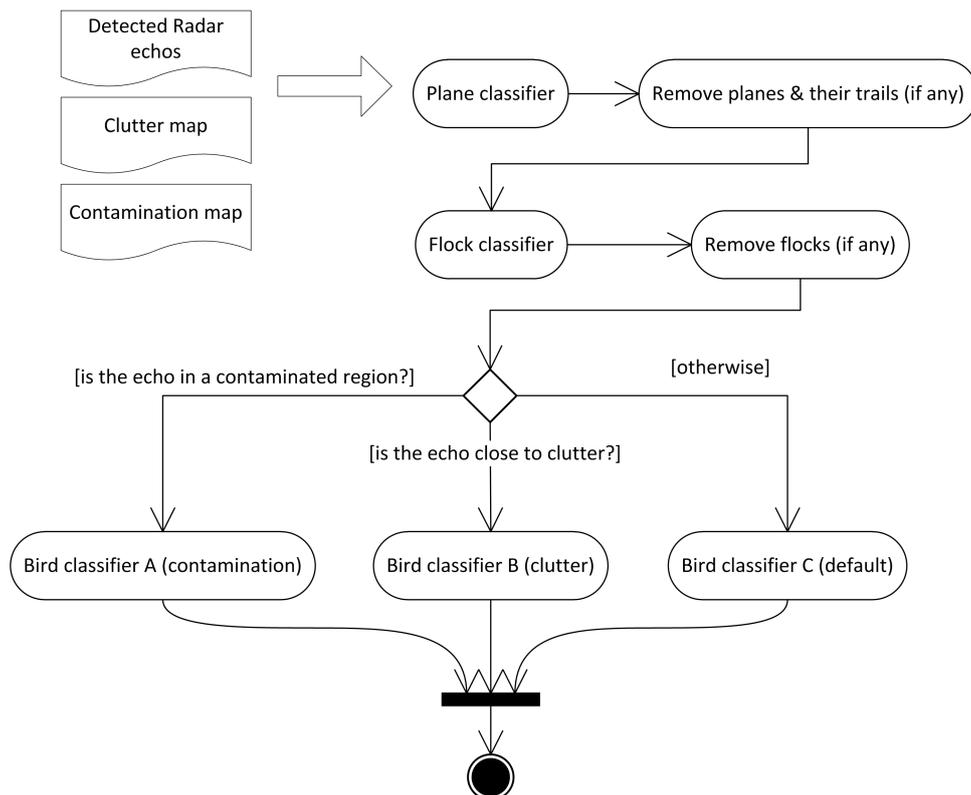

Fig. 2.17. Possible design of a multilevel bird classifier.

2.6.1 Support Vector Machines

The method chosen for the task of bird detection is the support vector machine (SVM), a classification algorithm based on optimal margins. The operating principle of support vector machines is conceptually straight forward, especially in the case of linearly separable classes where it consists of finding a hyperplane that maximizes the margin between two classes. Once a hyperplane is found, a classifier f which is able to predict the class $c \in \{-1, +1\}$ of any new observation x can be constructed

$$f(x) = \begin{cases} +1 \text{ if } w \circ x + b \geq 0 \\ -1 \text{ otherwise} \end{cases} \tag{2.25}$$

where w is the normal vector of the separating hyperplane and b is the intercept. The margin between classes (and with it the orientation of the decision surface) can be fully defined by a set of data points from each class. The job of the SVM training procedure is to identify these so called "support vectors" in the training data. In the two-dimensional case, maximizing the margin between classes can be thought of as finding a rectangle of maximum width which bisects the decision line and does not contain any data points. The margin of the SVM is the width of that rectangle [41, 42].
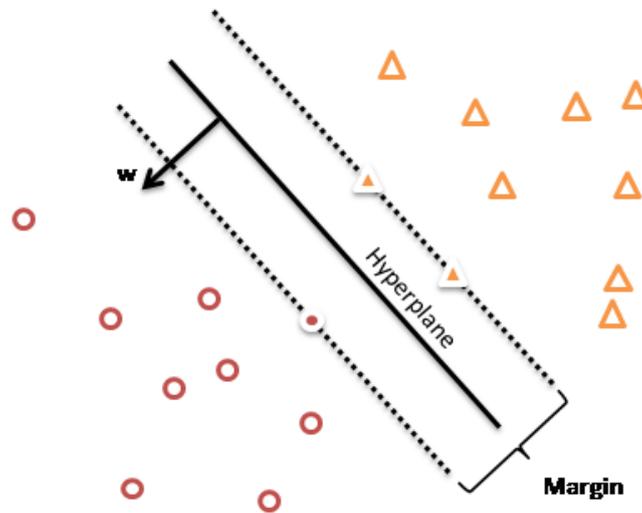
Fig. 2.18. SVM hyperplane separating two classes (represented by circles and triangles).

Finding an optimal hyperplane can be described as a constrained optimization problem which can be solved with quadratic programming techniques. y_l being the numeric class label for data point $x_l \in \{x_1, x_2, ..., x_m\}$ and α_l being the support vector coefficient for data point x_l, the training problem can be stated as

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i y_j \alpha_i \alpha_j k(x_i, x_j) - \sum_{i=1}^{m} \alpha_i \tag{2.26}$$

with constraints

$$\sum_{i=1}^{m} y_i \alpha_i = 0 \tag{2.27}$$

$$\alpha_i \geq 0, i = 1, 2, \ldots, m.$$

In general, the support vector coefficients α_l will be 0 for any point x_l not considered a support vector.

Obviously, in many real world applications, training sets will not be linearly separable. Introducing a kernel function k provides a convenient solution to this problem. A kernel function may be used to transform the data points to a new coordinate space in which they are easier to separate. Such kernel functions include the linear kernel (corresponding to the dot product)

$$k(s, t) = s \circ t, \tag{2.28}$$

the polynomial kernel of degree p

$$k(s, t) = (s \circ t + 1)^p, \tag{2.29}$$

and the Gaussian (radial basis function, RBF) kernel with free parameter σ

$$k(s, t) = \exp{-\frac{\|s - t\|^2}{2\sigma}}. \tag{2.30}$$

It will also often be the case, that none of the possible SVM models can perfectly separate the training set, resulting in some support vector coefficients growing to infinity. For this reason, slack variables are introduced in the SVM definition, which express themselves in an additional constraint for the quadratic programming problem. In practice, this means that a user defined "cost" value C is introduced, changing the constraint $\alpha_i \geq 0$ in equation 2.27 to $C \geq \alpha_i \geq 0$. The higher the value of C, the less errors are allowed, resulting in a smaller margin. As C decreases more errors are allowed, corresponding to a larger margin. Given a kernel function k and support vector coefficients α_i, the classifier f can now be expressed as

$$f(x) = \begin{cases} +1 \text{ if } \sum_{i=1}^{m} y_i \alpha_i k(x, x_i) - b \geq 0 \\ -1 \text{ otherwise.} \end{cases} \tag{2.31}$$

[41, 42]

SVM training and classification is performed using the `libsvm` package via the included Matlab interface [43].

When collecting training data, the fact that ground clutter is not uniform over extended time periods has to be considered. Therefore, if creating a representative training set consisting of images captured at different times, the respective clutter maps (which have been created individually for each source image series) need to be made available to the training procedure.[4]

---

[4] The software prototype created in the course of this project provides a simple user interface for creating training sets which include the corresponding clutter maps (see figure **??** in the appendix).

2.6.2 Classification Quality Measures

In order to assess the quality of the classification system, measures need to be chosen which adequately reflect the classifier's discrimination abilities. Because of the imbalanced nature of the data sets used here (significantly more non-bird echoes than bird echoes, even more so with plane echoes) the misclassification rate is not an adequate measure. For example, a classifier which always decides for the larger class may have an overall misclassification rate below 5% without classifying any element of the smaller class correctly. For this reason, other quality measures need to be considered.

Comparing the true class y_l of a sample x_l with its predicted class $f(x_l)$ can have four outcomes, which, for some testing set S_{test} can be summarized in a confusion matrix

$$
\begin{array}{cc|c}
TP & FN & 1 \\
FP & TN & -1 \\
\hline
1 & -1 &
\end{array}
\quad \text{true class}
$$

$$\text{predicted class}$$

where TP, TN, FP and FN stand for the number of true positives, true negatives, false positives and false negatives regarding class +1. Note that $TP + FN + FP + TN = |S_{test}|$.

Based on the confusion matrix, a number of classification quality measures can be computed. As the overall classification rate is not informative in case of imbalanced data sets, individual rates for the available classes should be calculated. The rate of correctly classified samples (sensitivity) for an examined class is defined as follows:

$$SN = \frac{TP}{TP + FN} \tag{2.32}$$

The analogous measure for the remaining class(es) is the specificity[5]

$$SP = \frac{TN}{TN + FP} \tag{2.33}$$

In order to assess the overall classification quality in a single scalar value, the geometric mean of the sensitivity and specificity measures may be used.

$$GM = \sqrt{SN \cdot SP} \tag{2.34}$$

An additional quality measure is provided by the correlation coefficient (also called Pearson or Matthews correlation coefficient in other contexts):

$$CC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FN) \cdot (TP + FP) \cdot (TN + FP) \cdot (TN + FN)}} \tag{2.35}$$

---

[5] Other definitions of sensitivity and specificity found in the literature include $SN = TP/(TP + FP)$ and $SP = TN/(TN + FN)$.

The correlation coefficient attains values between -1 (all decisions wrong) and $+1$ (all decisions correct). An absolute value close to one indicates a good classifier whereas a value close to zero indicates that the classifier performs no better than a system randomly assigning classes [44].

2.6.3 Classifier Visualization

Support vector machines, while providing high quality classification based on theoretically well-founded principles, are often used as a black box system. Classification quality is estimated based on accuracy measures for some training data set, while explaining why the SVM decides in a certain fashion is difficult. Nevertheless, in many applications, understanding how decisions are made are of grave importance (e.g. in medical applications). In the case of radar echo classification, it is of additional interest to find out on which echoes the classifier fails, to assess if there exists a regularity in the appearance of echoes being misclassified.

The most straightforward approach to SVM visualization is plotting the decision surface and the support vectors in two- or three-dimensional projections of the data. However, in most cases no analytic description of the decision surface is available. One workaround consists of creating a rectangular grid of points in the space defined by the value ranges of the input variables and classifying all of these sample points using the SVM. Those points where the SVM output is close to zero can be used as an approximation of the decision surface [45]. Obviously, this is a computationally demanding task subject to the curse of dimensionality and as such is confined to small numbers of input variables. If using variants of the SVM algorithm which give coefficient values of the separating hyperplane, such as the incremental SVM algorithm, the plane can be visualized rather conveniently [46]. A practical problem when visualizing the decision surface is, that although the surface may separate classes in higher dimensions, this separation is not necessarily visible in projected planes.

Several other approaches to SVM classifier visualization have been proposed, such as using nomograms ([47]) or self-organizing maps ([41]). However, the visualization method adopted here is displaying the training/testing data distribution in a histogram in which the bins correspond to the distances of samples to the decision surface of the SVM (see figure 2.19) [46, 48]. The distributions shown left of the histogram's ordinate axis contain those samples assigned to class -1 (here: *None*) whereas those samples classified as $+1$ (*Bird*) are included in the distributions to the right. The orientation of the bars reflects whether classifications were correct or not. Bars corresponding to true positives (true negatives) are shown pointing upward, false positives (false negatives) are shown pointing down. The histogram plot can also provide some information useful for tuning of SVM parameters. For example, if no samples are close to the class frontier, this suggests that at least one parameter (e.g. C, σ) has not been tuned finely [46].

In the created software prototype for bird detection, the bars of the histogram are additionally linked to the underlying samples in such a way that clicking a bar results in the involved radar echoes being selected and their feature values being

shown in a parallel coordinates plot. The echoes can then be individually viewed in a zoomed form and in the context of the source radar image (see figure **??** in the appendix).
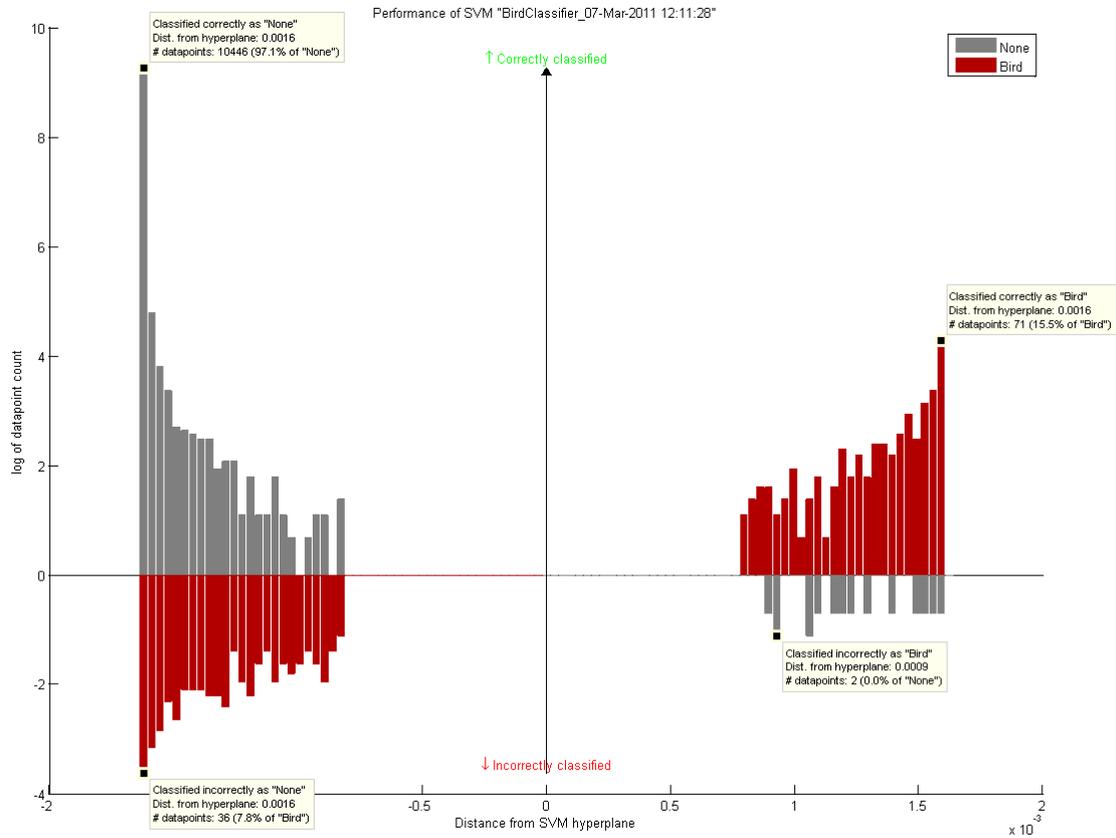

Fig. 2.19. SVM visualization histogram. Bars pointing upward represent correct decisions regarding the classes *Bird* (red) and *None* (gray), those pointing downward represent wrong decisions. Bar heights are logarithmic.

3

Results and Discussion

A Matlab prototype for bird detection in radar images has been developed. This prototype performs several preprocessing tasks, detection of ground clutter and insect contamination as well as echo segmentation and computation of echo features. Furthermore, means of inspecting feature distributions and training support vector machine classifiers for binary classification problems are provided.

The developed procedures for clutter and contamination map generation proved to be effective and stable throughout the encountered data and should be applicable almost seamlessly to images captured with different settings. There are however some free parameters of these methods, which might be optimized. In the future, seasonal information may be incorporated in the process, as insect contaminations are encountered more frequently in the summer months than in winter, when overall insect activity is close to zero.

As part of target segmentation and characterization, a trail identification method for the detected echoes was presented. The reasons for developing such a method were (1) that the images which this project is based on were captured at one minute intervals, making the trails plotted by the radar system the only source of temporal context information and (2) that plane trails need to be recognized in order to remove echoes produced along the plane trajectory. The method performs well, but there is room for improvement, as described in section 2.4.2. In general, it should be noted that if images captured at a higher rate are used, trail detection might not be as significant as it is with images captured once per minute, as then objects can be tracked by the software.

Unfortunately, no data sets with verified ground truth were available while conducting this project, thus complicating the development of a final classifier. Nevertheless, a multi-level classifier architecture has been proposed, based on the developed methods and the insights gained while working with the data. The outlined classification system deals with the numerous scenarios (no contamination, insect contamination, clutter, bird flocks, planes and plane trails) encountered in radar images by providing differently trained and configured classifiers for these situations.

With further development of the system, large and representative data sets will be created for training the individual classifiers, most likely requiring further assessment of feature relevancy regarding each classification task. This feature selec-

tion might be performed using the methods already incorporated in the prototype (PCA and SOM analysis, scatter plots and parallel coordinate plots) or by means of additional methods. Possibilities range from "brute-force" evaluation of SVM classification performance by training and testing with all feature combinations to more sophisticated feature selection approaches including filter methods (variable ranking independent of the choice of the predictor) and wrapper methods (using the chosen learning machine as a black box to score variable subsets according to their predictive power) [49].

For most of the time invested in this project, only images captured at one minute intervals were available. Towards the end of the project, a set of images captured at the maximum possible rate ("save-all-frames" mode of the frame grabber) were provided to us. Following visual inspection of these images, doubts arose, whether all of the assumptions made when the project was started were correct. As described above, the ground truth used in [27] and [28], which was supposed to form the basis for the methods developed in the scope of this project, was probably erroneous in that weak echoes were consistently eliminated or interpreted as insects. However, in the opinion of the authors of the present project, it is not possible to decide from individual images whether a radar echo corresponds to a bird or not, as targets detected in consecutive images exhibit substantial variation in appearance. Therefore, images captured at a higher frequency should be used, enabling the classification system to include temporal information in its decisions. Otherwise, a significant number of birds might be missed, because their aspect or position in the radar beam in the moment of detection caused a weak echo, or because they traversed the observed volume too quickly to be visible in the captured images. Should it after further consideration be deemed reasonable to perform extensive tracking in the new images, this will of course require a tracking approach to be chosen and implemented (e.g. multiple hypothesis tracking [50]).

As described above, the frame grabber's capturing frequency will in fact be significantly higher in the upcoming study at Munich airport. This was not decided as a consequence of the results presented here, but the afore-mentioned limitations of analyzing single images without temporal context stress the advantages of taking this step. Prior to conducting the study in Munich, it might be worthwhile to develop a small application performing on-the-fly PPI-GUI removal and cropping of radar images, with the intention of inserting it into the data flow process between the frame grabber and data storage. This would significantly reduce the storage space required for the large amounts of radar images produced at an increased capturing frequency.

Finally, it should be mentioned that the prototype presented here was not developed with efficiency in mind. When implementing a final system, optimizations need without question be applied in various spots.

References

1. R. A. Dolbeer, "Birds and aircraft - fighting for airspace in ever more crowded skies," *Human-Wildlife Conflicts*, vol. 3, no. 2, pp. 165–166, 2009.
2. N. S. Sodhi, "Competition in the air: Birds versus aircraft: Perspectives in ornithology," *The Auk*, vol. 119, no. 3, pp. 587–595, 2002.
3. D. Licht, "Statistik über die Vogelschläge der deutschen Zivilluftfahrt von 2007 bis 2008," *Vogel und Luftverkehr*, vol. 29, no. 1, pp. 1–6, 2009.
4. A. Borst, "Final report we@sea project robin lite bird radar development aimed at maritime bird migration monitoring," May 15, 2008.
5. "Fall 2005 mobile avian radar system (mars) monitoring report: Nantucket sound, massachusetts," October 2006.
6. S. A. Gauthreaux and C. G. Belser, "Overview: radar ornithology and biological conservation," *The Auk*, vol. 120, no. 2, pp. 266–277, 2003.
7. S. A. Gauthreaux and C. G. Belser, "Radar ornithology and the conservation of migratory birds: Technical report," 2005.
8. B. Bruderer, "The radar window to bird migration," in *Avian migration* (P. Berthold, ed.), pp. 347–358, Berlin: Springer, 2003.
9. B. Bruderer, "An introduction to radar ornithology," in *International Conference and Workshop on Radar Ornithology and Entomology, Helgoland (Germany)*, 2007.
10. F. Liechti and H. van Gasteren, "Current stage of bird radar systems," in *29. International Bird Strike Committee (IBSC) Meeting, Cairns (Australia)*, 2010.
11. A. Dokter, F. Liechti, L. T. P. Delobbe, H. Beekhuis, and I. Holleman, "Bird detection by operational weather radar," in *Proc. 6th European Conf. on Radar in Meteorology and Hydrology: Adv. in Radar Technology, Sibiu, Romania*, 2010.
12. DeTect Inc., "Detect merlin: Avian radar systems: Werbebroschüre," 2007.
13. T. J. Nohara, P. Weber, A. Premji, C. Krasnor, S. Gauthreaux, M. Brand, and G. Key, "Affordable avian radar surveillance systems for natural resource management and bash applications," in *2005 IEEE International Radar Conference record*, (Piscataway, NJ), pp. 10–15, IEEE Service Center, 2005.
14. A. Borst, "Robin lite low cost mobile 3d bird radar."

15. R. A. Dolbeer, S. E. Wright, J. Weller, and M. J. Begier, "Wildlife strikes to civil aircraft in the united states 1990-2008: National wildlife strike database serial report number 15," Washington, DC, September 2009.

16. J. Thorpe, "Update on fatalities and destroyed civil aircraft due to bird strikes with appendix for 2008 & 2009," in *29. International Bird Strike Committee (IBSC) Meeting, Cairns (Australia)*, 2010.

17. W. J. Richardson and T. West, "Serious birdstrike accidents to military aircraft of many countries: Additions and known totals," in *26. International Bird Strike Committee (IBSC) Meeting, Warsaw*, 2003.

18. S. C. Barras and T. W. Seamans, "Habitat management approaches for reducing wildlife use of airfields," in *Proceedings of the Vertebrate Pest Conference* (R. M. Timm and R. H. Schmidt, eds.), pp. 309–315, 2002.

19. A. Dekker, "Taking habitat management one step further," in *26. International Bird Strike Committee (IBSC) Meeting, Warsaw*, 2003.

20. International Birdstrike Committee, "Standards for aerodrome bird/wildlife control," October 2006.

21. A. Ludloff, *Handbuch Radar und Radarsignalverarbeitung*. Braunschweig/Wiesbaden: Vieweg, 1993.

22. F. Liechti, "Combining radar systems to get a 3d-picture of the bird migration," in *28. International Bird Strike Committee (IBSC) Meeting, Brasilia*, 2008.

23. S. Zaugg, G. Saporta, E. van Loon, H. Schmaljohann, and F. Liechti, "Automatic identification of bird targets with radar via patterns produced by wing flapping," *Journal of the Royal Society, Interface / the Royal Society*, vol. 5, no. 26, pp. 1041–1053, 2008.

24. H. Schmaljohann, F. Liechti, E. Bächler, T. Steuri, and B. Bruderer, "Quantification of bird migration by radar – a detection probability problem," *Ibis*, vol. 150, no. 2, pp. 342–355, 2008.

25. E. E. Herricks, E. Woodworth, and R. King, "Deployment of avian radars at civil airports," February 2010.

26. P. Weber, T. J. Nohara, and S. Gauthreaux, "Affordable, real-time, 3-d avian radar networks for centralized north american bird advisory systems," in *Bird Strike 2005 Conference*, 2005.

27. M. Deutsch, "Untersuchungen zum Vogelaufkommen am Flughafen Leipzig-Halle: Endbericht: Durchgeführt für die Flughafen Leipzig/Halle Gmbh im Rahmen des Planfeststellungsverfahren," Jun. 2009.

28. G. Kuschk, *Automatische Vogelzug-Erfassung mittels Radarbildern*. PhD thesis, Martin-Luther-Universität, Halle-Wittenberg, 15. Juli 2007.

29. B. Bruderer, "Zur Registrierung und Interpretation von Echosignaturen an einem 3-cm-Zielverfolgungsradar," *Der Ornithologische Beobachter*, no. 66, pp. 70–88, 1969.

30. B. Bruderer, "Short-range high-precision surveillance of nocturnal migration and tracking of single targets," *Israel Journal of Zoology*, vol. 41, pp. 207–220, 1995.

31. "TONI Vogelschlag-Prävention: Werbebroschüre," Frankfurt am Main, 2010.

32. G. J. Awcock and R. Thomas, *Applied image processing.* New York: McGraw-Hill, new ed. ed., 1996.

33. P. J. Clark and F. C. Evans, "Distance to nearest neighbor as a measure of spatial relationships in populations," *Ecology*, vol. 35, no. 4, pp. 445–453, 1954.

34. W. Bunch and E. E. Herricks, "Observational validation of avian radar systems," in *FAA Worldwide Airport Technology Transfer Conference*, 2010.

35. J. R. Parker, *Algorithms for image processing and computer vision.* Indianapolis, Ind.: Wiley, 2. ed. ed., 2011.

36. MATLAB, *Image Processing Toolbox Documentation (version 7.1).* Natick, Massachusetts: The MathWorks Inc., 2010.

37. D. Lefkaditis and G. Tsirigotis, "Morphological feature selection and neural classification for electronic components," *Journal of Engineering Science and Technology Review*, no. 1, pp. 151–156, 2009.

38. I. T. Jolliffe, *Principal Component Analysis.* Springer series in statistics, New York: Springer, 2. ed. ed., 2002.

39. H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdisciplinary Reviews Computational Statistics*, vol. 2, no. 4, pp. 433–459, 2010.

40. J. Vesanto, J. Himberg, E. Alhoniemi, and J. Parhankangas, "Self-organizing map in matlab: the som toolbox," in *Proceedings of the Matlab DSP Conference*, pp. 35–40, 2000.

41. L. Hamel, "Visualization of support vector machines with unsupervised learning," in *IEEE Symposium on Computational Intelligence and Bioinformatics and Computational Biology*, 2006.

42. M. P. Wand, "Support vector machine classification," *Parabola*, vol. 42, no. 2, 2006.

43. C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines: Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm," 2001.

44. P. Baldi, S. Brunak, Y. Chauvin, C. A. F. Andersen, and H. Nielsen, "Assessing the accuracy of prediction algorithms for classification: an overview," *Bioinformatics*, vol. 16, no. 5, pp. 412–424, 2000.

45. D. Caragea, D. Cook, H. Wickham, and V. G. Honavar, "Visual methods for examining svm classifiers," in *Visual Data Mining* (S. J. Simoff, M. H. Böhlen, and A. Mazeika, eds.), pp. 136–153, Springer-Verlag, 2008.

46. F. Poulet, "Towards effective visual data mining with cooperative approaches," in *Visual Data Mining* (S. J. Simoff, M. H. Böhlen, and A. Mazeika, eds.), vol. 4404, pp. 389–406, Springer-Verlag, 2008.

47. B. H. Cho, H. Yu, J. Lee, Y. J. Chee, I. Y. Kim, and S. I. Kim, "Nonlinear support vector machine visualization for risk factor analysis using nomograms and localized radial basis function kernels," *IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 12, no. 2, pp. 247–256, 2008.

48. F. Poulet, "Svm and graphical algorithms: A cooperative approach," in *Proceedings of the Fourth IEEE International Conference on Data Mining* (R. Rastogi, ed.), pp. 499–502, Los Alamitos, Calif.: IEEE Computer Society, 2004.

49. I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

50. S. S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 19, no. 1, pp. 5–18, 2004.