

Dynamic Programming and Skyline Extraction in Catadioptric Infrared Images

Jean-Charles Bazin, Inso Kweon, Cédric Demonceaux, Pascal Vasseur

▶ **To cite this version:**

Jean-Charles Bazin, Inso Kweon, Cédric Demonceaux, Pascal Vasseur. Dynamic Programming and Skyline Extraction in Catadioptric Infrared Images. IEEE International Conference on Robotics and Automation - ICRA, May 2009, Kobe, Japan. hal-01779780

HAL Id: hal-01779780
https://hal.science/hal-01779780

Submitted on 26 Apr 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Dynamic Programming and Skyline Extraction in Catadioptric Infrared Images

Jean-Charles Bazin, Inso Kweon, Cédric Demonceaux and Pascal Vasseur

Abstract— **Unmanned Aerial Vehicles (UAV) are the subject of an increasing interest in many applications and a key requirement for autonomous navigation is the attitude/position stabilization of the vehicle. Some previous works have suggested using catadioptric vision, instead of traditional perspective cameras, in order to gather much more information from the environment and therefore improve the robustness of the UAV attitude/position estimation. This paper belongs to a series of recent publications of our research group concerning catadioptric vision for UAVs. Currently, we focus on the extraction of skyline in catadioptric images since it provides important information about the attitude/position of the UAV. For example, the DEM-based methods can match the extracted skyline with a Digital Elevation Map (DEM) by process of registration, which permits to estimate the attitude and the position of the camera. Like any standard cameras, catadioptric systems cannot work in low luminosity situations because they are based on visible light. To overcome this important limitation, in this paper, we propose using a catadioptric infrared camera and extending one of our methods of skyline detection towards catadioptric infrared images. The task of extracting the best skyline in images is usually converted in an energy minimization problem that can be solved by dynamic programming. The major contribution of this paper is the extension of dynamic programming for catadioptric images using an adapted neighborhood and an appropriate scanning direction. Finally, we present some experimental results to demonstrate the validity of our approach.**

I. INTRODUCTION

Estimating the position and the orientation of a Unmanned Aerial Vehicle (UAV) is a fundamental step towards the autonomy of these vehicles. It is now well established that traditional navigation equipments like Global Positioning System or Inertial Navigation System suffer from many limitations. For example, GPS is sensitive to signal dropout and hostile jamming. The drawback of INS is that its position error compounds over time and may cause large localization errors. In order to overcome these main disadvantages, many researchers suggested a vision-based approach of the navigation problem, which helps estimating localization and/or orientation of a UAV when GPS or inertial guidance is not available [1][2][3][4]. Most of the existing works use conventional cameras which have a narrow field of view and thus the amount of information we could get from the environment is drastically limited. That is why some works have proposed using catadioptric cameras to greatly increase the field of view and applied them for various tasks [5][6][7].

J.C. Bazin and I.S. Kweon are with RCV Lab, EE department, KAIST, South Korea. jcbazin@rcv.kaist.ac.kr and iskweon@kaist.ac.kr

C. Demonceaux and P. Vasseur are with MIS laboratory, Université of Picardie Jules Verne, France. (cedric.demonceaux, pascal.vasseur)@u-picardie.fr

Concerning UAV applications, Hrabar and Sukhatme were the first to apply catadioptric cameras for autonomous aerial vehicles. In [8][9], they managed to estimate the 2D position/orientation of an helicopter. However, they require artificial visual targets and cannot estimate the 3D position/orientation. Our research group has been aiming to estimate the UAV attitude/position, which has led to a series of recent publications. The methods introduced in [10] and [11] compute the roll and pitch angles after extracting the horizon in catadioptric images by adapted Markov Random Field (MRF) or maximizing RGB-based Mahalanobis distance. We have also developed a line-based method that can run in urban environment [12]. By considering the vertical direction as the normal vector of the ground plane, we were able to apply the same technique than in [10][11]. In [13], we extracted the vanishing points contained in an urban scene and managed to estimate the complete rotation using infinite homography. In [14], we developed a top-down approach that builds some hypothesis on the rotation angles and then verifies the consistency of the vanishing points extraction. In [15], we developed a hybrid approach that combines horizon-based technique with planar homography in order to estimate both rotation and translation. Finally, in [16], we have introduced a method that combines rotation estimation and epipolar geometry to estimate the attitude and the position of a UAV from only 2 point correspondences.

Whereas very interesting results have been obtained, all these vision-based methods cannot operate in low luminosity conditions since cameras are generally based on visible light which limit their applications to daytime and appropriate weather conditions. The solution is to use an infrared camera, which permits to observe the environment independently of the luminosity conditions. Unfortunately, most of the navigation methods developed for traditional images cannot be applied for IR images since traditional and IR images contain different types of information. This is essentially due to the observability problem: the features that are generally used for navigation estimation cannot be correctly detected in IR images. For example, the lines associated to urban environment (building edges, road lanes, etc...) are hard to extracted because they usually do not emit distinctive IR radiations and thus the line-based methods are likely to fail [13]. The spatial resolution of IR images is usually very low and moreover, in UAV applications, since the camera is at high altitude, the image resolution might still be lower (1 pixel represents a larger distance). Therefore the extraction of interest points, like SIFT features [17] or Harris corners [18], might not be precise and most of the traditional methods

based on epipolar geometry cannot work properly [19].

In this paper, we aim to extract the skyline in catadioptric infrared images since it provides key information about the attitude/position of the UAV. For example, once the skyline is extracted, the DEM-based methods can match it with a Digital Elevation Map (DEM) by process of registration [2][20], which permits to estimate the attitude and the position of the camera. Due to the narrow field of view of traditional cameras, only a small part of the skyline is usually observed and thus the matching step with the DEM is not very robust. By using a catadioptric infrared camera, we can observe, independently of the luminosity conditions, a much larger part of the skyline, which can improve the registration with the DEM. The task of extracting the best skyline in images is usually converted in an energy maximization problem that can be solved by the important technique of dynamic programming. However the traditional dynamic programming is not adapted for catadioptric images because of the distortions due to the mirror. The major contribution of this paper is the extension of the dynamic programming technique for catadioptric images using an adapted neighborhood and an appropriate scanning direction.

This paper is divided into four main parts. First, we will recall the catadioptric projection and explain the concept of the catadioptric infrared cameras. In the third part, we present our proposed algorithm for skyline extraction in catadioptric IR images and the extension of dynamic programming. Finally we perform some experiments and show the results.

II. CATADIOPTRIC VISION AND IMAGE FORMATION

Intuitively, if the field of view is wider, we can gather more information from the environment and thus it is possible to estimate the pose more precisely and robustly. Obviously, an imaging system that is able to see in all directions could play a key role for this task. Such kind of sensors is simply called omnidirectional systems and provides a wide field of view. Catadioptric cameras are a specific kind of omnidirectional systems. They are devices which use both mirrors (catoptric elements) and lenses (dioptric elements) to form images through a conventional camera [21]. Such systems usually have a field of view greater than 180 degrees and are getting both cheaper and more effective. Baker and Nayar classified catadioptric sensors into two categories depending on the number of viewpoints [22]. Sensors with a single viewpoint, named central catadioptric sensors, permit a geometrically corrected reconstruction of the perspective image from the original catadioptric image. Geyer and Daniilidis have demonstrated the equivalence for the single viewpoint category with a two-step projection via a unitary sphere centered on the focus of the mirror (the single viewpoint) [21]. This two-step projection consists first in projecting a real 3D point P_w to a point P_s from the center of the sphere O_c (Fig 1). The second step projects point P_s to point P_i in the image plane from a specific point O_p. In order to apply the equivalence, it is necessary to know the intrinsic parameters of the camera and two additional parameters namely ξ and φ which are respectively equal to distances $|O_cO_p|$ and $|O_cO_i|$. Parameters ξ and φ define the shape of the mirror and can be estimated by calibration [23].

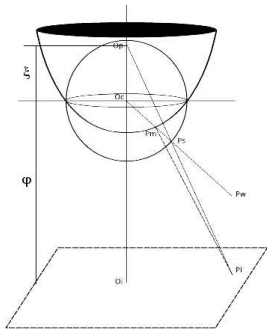


Fig. 1. Equivalence between the catadioptric projection and the two-step mapping via the sphere

III. CATADIOPTRIC INFRARED CAMERA

When a scene contains a very low luminosity, the images acquired by standard cameras are almost black because these cameras are based on visible light. To solve this problem, infrared cameras have been developed. They are based on infrared (IR) radiation which is an electromagnetic radiation of a wavelength longer than that of visible light, i.e. IR rays cannot be seen by humans. The interesting property is that objects emit IR radiations depending on their temperature and the composition of their material, independently of the luminosity in the scene. Therefore, even in completely dark situations, the radiation of objects can be detected and we can visualize the difference of IR radiations among objects.

The difficulty of developing an infrared catadioptric camera is that the catadioptric mirror must correctly reflect the IR rays of the scene objects. Indeed, if this requirement is not fulfilled, then the IR image will mainly correspond to the IR rays emitted only from the mirror, instead of those from the scene objects. As a consequence, the acquired image will be quasi homogeneous and the scene objects cannot be perceived in the image. The solution of this problem consists in coating the mirror with an IR-reflective layer so that the IR rays can be correctly reflected by the mirror and then captured by the IR camera. Indeed, mirror surfaces can be overcoated both to retard degradation of the surface and to increase their reflectivity in parts of the spectrum where they will be used, i.e. infrared spectrum for our specific application. An important property of the so-called *hot mirror* is that they are transmissive to visible light but reflective to infrared light. In a manufacturing point of view, it consists of a thin-film decomposition method for which common techniques exist (vapor deposition, beam deposition, etc...).

Compared to traditional IR cameras, catadioptric infrared systems could cover a much larger part of the scene. They could usually permit to observe at 360 degrees in the horizontal plane and around 180 degrees in the vertical plane. If one wants to cover the same field of view with a network

of several traditional IR cameras, it will be both much more expensive and inconvenient (synchronization, weight, price, etc...).

IV. SKYLINE EXTRACTION

A. Perspective IR images

In [24], neuroscientists discovered that information contained in an image (color, texture, shape, etc...) is first analyzed by distinct neurons and then integrated into one object. These two steps are respectively called the attention and the binding processes. Figure 2 depicts a typical example of an IR image obtained by a traditional perspective IR camera. It can be noticed that the skyline is a strong edge boundary which separates the sky and mountain regions. To extract the skyline, the human visual system focuses on two elements (edgeness and homogeneity) during the attention process and then gather information during binding process. Therefore based on this biological evidence of the human vision system, we defined in [25] the energy of a pixel (x, y) as:

$$E(x,y) = \alpha_1 |\nabla G_\sigma * I(x,y)| + \alpha_2 \frac{D(x,y)}{H} \qquad (1)$$

This energy corresponds to the binding process and is composed of two terms. The first one corresponds to edgeness: the magnitude of the convolution between edge operator G_σ (in horizontal and vertical directions and where σ is the standard deviation of the Gaussian distribution) and the image I at pixel (x, y). The second term represents the homogeneity. For perspective images, it was computed by the number $D(x, y)$ of pixels of similar intensity in the upper and lower directions, where upper and lower regions must have different mean intensities (to represent sky and mountain regions). This distance $D(x, y)$ is then normalized by the image height H. α_1 and α_2 correspond to the weights associated to the edgeness and homogeneity terms. If a pixel has a high energy, then it is considered a strong candidate pixel on the skyline.

Finally, the skyline is extracted by finding the path whose total energy is maximum. This energy maximization problem can be solved efficiently by dynamic programming [26]. Dynamic programming (DP) is a method used to optimize some problems containing optimal substructure, such as finding the best path between two nodes in a graph. Optimal substructure means that optimal solutions of subproblems can be used to find the optimal solution of the overall problem. DP is usually composed of 3 main steps: initialization, scoring and back-tracing. In the application of finding the best skyline in traditional IR images, the initialization step creates a matrix Q whose size is $H \times W$ where H and W respectively correspond to the image height and width. Once the energy $E(x, y)$ of every pixel is computed using eq (1), the scoring step fills in a matrix Q such that $Q(x, y)$ is the highest energy to reach the pixel $P = (x, y)$ from any point on the first column (i.e. $x = 1$). For our application, the scoring can be performed recursively as followed:

$$Q(x,y) = \begin{cases} E(x,y) \text{ if } x = 1 \\ \min(Q(x-1,y-1), Q(x-1,y), Q(x-1,y+1)) \\ +E(x,y) \text{ otherwise} \end{cases} \qquad (2)$$

Note that this recursive form is in forward direction, in the sense that the image is scanned by increasing the x value. The third and final step of DP finds the path that maximizes the total energy by tracing backward the matrix Q. This can be done easily by using a predecessor array that saves where the best path come from in the recursive algorithm of eq (2). Thus, in our application, we can obtain the path that most probably corresponds to the skyline. It is worthwhile noting that if wide occlusion occurs (e.g. at low altitude in urban environment, a large building might occlude the skyline), then obviously the true skyline cannot be recovered, but the DP will still extract the path that has the highest energy, even if some parts of the path might not correspond to the true skyline.

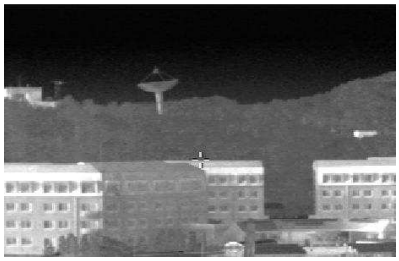


Fig. 2. Typical example of an IR image obtained by a traditional perspective IR camera.

B. Extension of Dynamic Programming for Catadioptric IR Images

Unfortunately it is not possible to apply this skyline extraction algorithm directly in catadioptric images. For example, in the perspective case, we simply apply the dynamic programming technique column-wisely but obviously, this is not valid for catadioptric images. Moreover the notion of neighborhood is different in catadioptric images due to the mirror distortion. The basic solution would be to unwarp the catadioptric image into a panoramic image and then apply an existing skyline extraction method. However

this approach suffers from some limitations. Because of interpolation, the unwarping process is very time consuming and also introduces extra noise into the image. Moreover on the rectified image, there still exist underlying distortions. That is why we propose to redefine dynamic programming for catadioptric images. First of all, DP strongly relies on the notion of neighborhood. For example, in eq (2), we check the energy of the pixels (x', y') that are in the neighborhood of the current pixel (x, y) and whose horizontal coordinate is smaller than the one of the current pixel (i.e. *forward direction*: $x' < x$). The neighborhood was simply defined as the 3×3 square centered on the current pixel. However, because of the distortions inherent to the mirror, this neighborhood definition is not valid in catadioptric images and has to be modified. Intuitively, the size of the neighborhood of a pixel in a catadioptric image depends on the position of this pixel in the image: if it lies at the center of the image, the neighborhood is smaller than if it lies at the image boundary. Thus it is complicated to work with neighborhoods in catadioptric images. In order to deal with neighborhoods more easily and define the dynamic programming technique in catadioptric images, we propose working in the equivalent sphere space rather than on the 2D image. The sphere space greatly simplifies the formalism to take into account the distortions due to the mirror. Each point $P_i = (x, y)$ in the image plane I^2 can be projected onto the equivalent sphere S^2 at $P_s = (\theta, \phi, r)$, where $\theta \in [-\pi; \pi]$ (azimuth angle), $\phi \in [-\pi/2; \pi/2]$ (elevation angle) and the radius $r = 1$. In sphere space, the distance between two points can be simply measured using the geodesic length. Concretely, the angular distance between two spherical points P_s^1 and P_s^2 is calculated by $acos(P_s^1 \cdot P_s^2)$. We note $\Lambda()$ the function that projects an image point into the sphere, i.e. $P_s = \Lambda(P_i)$. In [27], we proposed to define the spherical neighborhood $N_s(P_s)$ of a spherical point $P_s = (\theta, \phi, 1)$ as:

$$N_s(P_s) = \left\{ \begin{array}{l} P'_s = (\theta', \phi', 1) \in S^2 \text{ such that} \\ P'_s \neq P_s \text{ and} \\ |\phi - \phi'| \leq \phi_{thresh} \text{ and} \\ \min(|\theta - \theta'|, 2\pi - |\theta - \theta'|) \leq \theta_{thresh} \end{array} \right. \quad (3)$$

In plain English, the neighborhood $N_s(P_s)$ is the set of spherical points contained in a patch centered at P_s and whose lengths along θ and ϕ directions are respectively θ_{thresh} and ϕ_{thresh}. As in [27], the neighborhood $N_i(P_i)$ of a point P_i in the image plane I^2 is obtained by projecting the neighborhood of its spherical point onto the image plane:

$$N_i(P_i) = \{P'_i \in I^2 / \Lambda(P'_i) \in N(\Lambda(P_i))\} \quad (4)$$

For notation, the subscripts of N_i and N_s are used to emphasize that N_i and N_s respectively corresponds to the neighborhoods in the image plane and the sphere space. Using spherical coordinates, an intuitive extension of the forward direction (i.e. $x' < x$) for catadioptric images can simply be $\theta' < \theta$, which we refer as the θ *forward direction*. Therefore, given a point P_i, we note $\widetilde{N_i}(P_i)$ the set of image points that correspond to the neighborhood predecessors of P_i in the θ forward direction. Mathematically, we can define it as:

$$\widetilde{N_i}(P_i) = \left\{ \begin{array}{l} P'_i \in I^2 \\ \text{ such that } P'_s \in N(P_s) \text{ and } \theta' < \theta \\ \text{ where } P_s = \Lambda(P_i) = (\theta, \phi, 1) \\ \text{ and } P'_s = \Lambda(P'_i) = (\theta', \phi', 1) \end{array} \right. \quad (5)$$

Now that we can correctly scan the catadioptric image with an adapted neighborhood and an appropriate forward direction, we can finally apply the equivalent of the recursive function (2) to compute the best path to reach any point P_i:

$$Q(P_i) = \left\{ \begin{array}{l} E(P_i) \text{ if } \Lambda(P_i) = (-\pi, \phi, 1) \\ \min_{P'_i \in \widetilde{N_i}(P_i)} Q(P'_i) + E(P_i) \text{ otherwise} \end{array} \right. \quad (6)$$

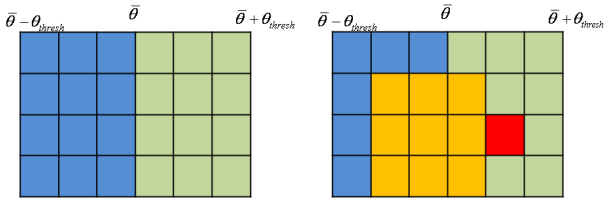
C. Implementation Issues of Dynamic Programming

As previously explained, we implement the dynamic programming for catadioptric images in the θ forward direction. Let suppose that the scoring matrix Q has been filled in up to a particular $\overline{\theta}$, i.e. all the pixels whose spherical projection (θ, ϕ, r) is such that $\theta \leq \overline{\theta}$ have been scanned. To continue filling in the score matrix, two tasks must be performed. First, we need to scan the pixels that are near the previously scanned pixels and in the forward direction (in perspective case, it corresponds to the points in the next column). Mathematically, this is the set of pixels P'_i where $\Lambda(P'_i) = (\theta', \phi', 1)$ is such that $\theta' > \overline{\theta}$ (forward direction) and $\theta' - \overline{\theta} < \theta_{thresh}$ (continuity). The second task consists in getting the predecessors of these pixels in order to apply the recursive algorithm of eq (6).

At each iteration, $\overline{\theta}$ is increased by θ_{thresh} (in perspective case, it corresponds to the next column, i.e. $x = x+1$). Thus at the i^{th} iteration, the θ angles of the current spherical points belong to the interval $[-\pi + (i-1)\theta_{thresh} : -\pi + i\theta_{thresh}]$. As this θ interval is determined in advance, an efficient implementation for the first task consists in sorting the spherical points with respect to their θ angles, which permits to access the spherical points belonging to some bounded θ intervals and their 2D image coordinates very easily, and then save the results in a LookUpTable in pre-processing.

Therefore, at any iteration, we can calculate the current θ interval and obtain the list of the pixels belonging to this θ interval instantaneously.

For the second task, we want to quickly obtain the neighborhood predecessors $\widetilde{N}_i(P_i)$ of any point P_i. A basic but efficient solution is to build a second LookUpTable which contains the list of predecessors for every point P_i. However some predecessors of a given point P_i might lie in the same θ interval than P_i (cf Fig 3). This is an important problem. Indeed, in this case, the path length (i.e. the number of pixels) to go from a θ interval to another is not fixed. If a path is longer, its energy is likely to be higher since more energies will be summed up and as a consequence, DP might extract long paths that do not correspond to the expected skyline. Therefore these predecessors must not be used when computing the path score. To simply solve this problem, in our implementation, we have simply removed the points of $\widetilde{N}(P_i)$ that lie in the same θ interval than the current pixel and save the results in a LookUpTable in preprocessing.



(a) (b)

Fig. 3. Scanning in the θ forward direction. For an easier interpretation, the deformed grid (θ, ϕ) of a catadioptric image is represented as a uniform and orthogonal grid. In (a), the image is split along θ into 2 regions: one where $\theta \in [\bar{\theta}-\theta_{thresh} : \bar{\theta}]$ (in blue) and one where $\theta \in [\bar{\theta} : \bar{\theta}+\theta_{thresh}]$. In (b), given a current pixel P_i (drawn in red), we can get its predecessors $\widetilde{N}_i(P_i)$ (drawn in orange) but some of them might lie in the same θ interval than the current pixel P_i.

The way of splitting the sphere into fixed θ intervals can also be used to define some upper and lower directions. Each θ interval can be considered as a vertical slice, as depicted in Fig 4. The upper and lower directions can be scanned by increasing and decreasing the ϕ angle respectively in a particular vertical slice. Thus we are also able to redefine the homogeneity term of eq (1) which can be calculated by Algorithm 1. Starting from a point P in the image, we scan the pixels belonging to the same θ range that P in upper and lower directions until the pixel intensity differs significantly from the intensity of P. Then, we compute the mean intensities of the two upper and lower regions and we measure how different the mean intensities are.

Algorithm 1 $distance$=FnComputeHomogeneity(P)

// $P = (x, y)$: point in the 2D image I
// L_{θ_r} : list of pixels belonging to the theta range θ_r and sorted with respect to ϕ angle in ascending order
$\theta_r = FnGetThetaRange(P)$
$k^{ori} = FnGetIndex(P, L)$ *// get the index of P in list L*
$\phi^{ori} = FnGetPhi(P)$ *// get the ϕ angle of P*

// similarity in upper direction
$k = k^{ori}$
repeat
  $k = k + 1$
until $P' = L_{\theta_r}(k)$ such that $|I(P) - I(P')| > Thresh_1$
$dist_{up} = (k-1) - k^{ori}$
$$Value_{up} = \sum_{k'=k^{ori}}^{k-1} I(L_{\theta_r}(k'))$$

// similarity in lower direction
$k = k^{ori}$
repeat
  $k = k - 1$
until $P' = L_{\theta_r}(k)$ such that $|I(P) - I(P')| > Thresh_1$
$dist_{down} = k^{ori} - (k+1)$
$$Value_{down} = \sum_{k'=k+1}^{k^{ori}} I(L_{\theta_r}(k'))$$

// homogeneity
if $\left|\frac{Value_{up}}{dist_{up}} - \frac{Value_{down}}{dist_{down}}\right| > Thresh_2$ **then**
  $distance = dist_{up} + dist_{down}$
else
  $distance = 0$
end if

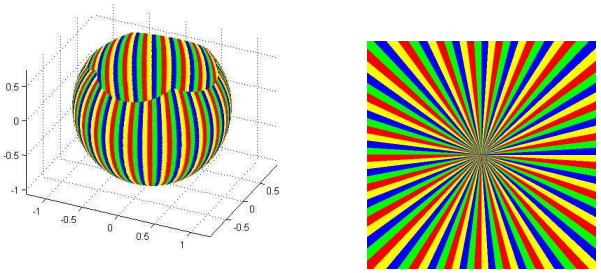


Fig. 4. By dividing the range of the θ angles into several intervals, we can obtain some vertical slices in the equivalent 3D sphere (left) and the associated 2D image (right). For a better visualization, the θ range has been split into 100 intervals only.

V. EXPERIMENTAL RESULTS

A. Synthesizing Catadioptric IR Images

To date, IR-reflective catadioptric mirrors are not commercially available yet. Thus it is impossible for us to purchase this equipment. Nevertheless, in order to show the validity of our approach, we performed two important tasks. First of all, to demonstrate the properties of IR-reflective mirror, we have combined an infrared camera with a planar IR-reflective mirror. We placed some objects in front of the mirror and pointed the IR camera towards the mirror. As expected, the IR camera was able to correctly visualize the reflection of the object on the IR-reflective mirror despite the complete darkness in the room.

In the second task, we decided to synthesize some catadioptric IR images in order to demonstrate the validity of our skyline extraction and our adapted dynamic programming. An approach to generate some synthesized catadioptric IR images would be to construct a virtual world and assign a material composition and a temperature value to each object of the world [28][29]. However the development would take a very long time and require deep knowledge in graphics and physics. That is why we preferred exploring another approach. Our idea consists in building a catadioptric IR image from a real perspective IR image. For this, we propose an algorithm composed of 5 main steps. Fig 5 depicts the whole process and the final result. Some extra results are also presented in Fig 8. This approach provides two key advantages: first, it is simple to develop and second, it verifies not only the physical properties of a real IR camera (since we are working on a real IR image) but also the geometrical properties of a catadioptric camera (since we used a central catadioptric projection model with synthesized calibration parameters).

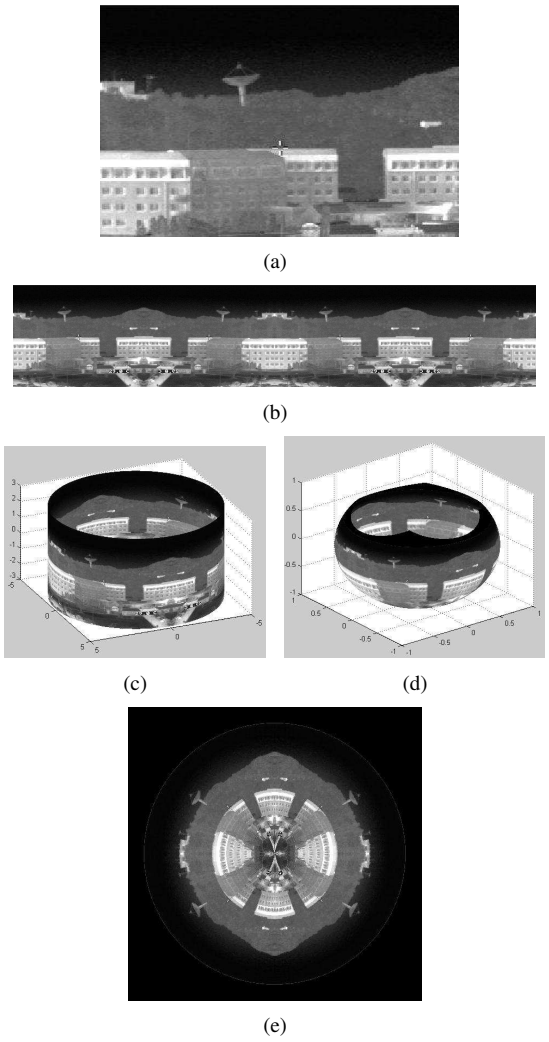


Fig. 5. The five steps to synthesize a catadioptric IR image from a real perspective IR image. Step1: get a real perspective IR image (a), step2: build a panoramic image by repeating the original image (b), step3: convert to a cylindrical representation (c), step4: project into the equivalent sphere (d), step5: project on the 2D image plane (e).

B. Skyline Extraction

This section presents some experimental results of skyline extraction in catadioptric infrared images. To obtain a dense grid of the catadioptric image (i.e. make the θ intervals thinner), we split the ranges of the θ angles into 500 intervals. We set the weights α_1 and α_2 of eq (1) both to 1 (equal weight). Figure 6 represents the distribution of the strong skyline candidates, i.e. whose energy is higher than a threshold $\mu = 1.05$ in our experiments. Figure 7 depicts the final result of skyline extraction obtained by our adaptation of the dynamic programming. Figure 8 shows extra results obtained for another acquisition. To show that our method also works when the skyline is not entirely visible, we synthesized some camera rotation so that the skyline is partially occluded and we manually fixed the limits of the θ angle. Results are presented in Fig 9. These experimental results demonstrate the validity of our approach.

VI. CONCLUSION

In this paper, we aimed to extract the skyline in catadioptric infrared images. The skyline can provide important information for aerial robotic applications. For example, the DEM-based methods can match the skyline with a Digital Elevation Map (DEM) by process of registration and then estimate the orientation/position of the camera.

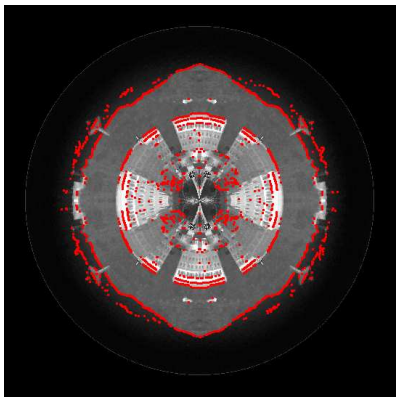


Fig. 6. Distribution of horixels

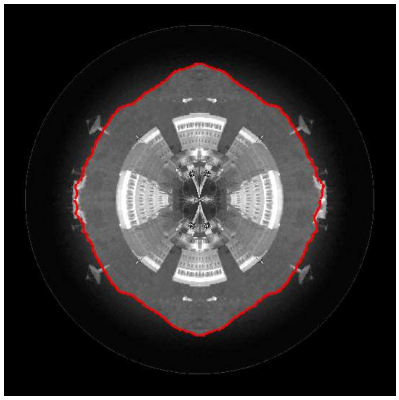


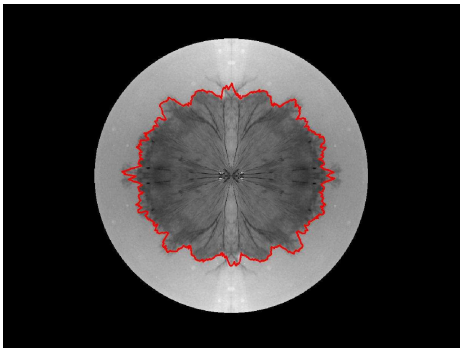
Fig. 7. Skyline extraction by our adapted dynamic programming



(a)



(b)



(c)

Fig. 8. Results of skyline extraction for an other acquisition: real perspective IR image (a), synthesized panoramic image (b), synthesized catadioptric image and skyline extraction (c).

In order to overcome the limitations of traditional vision systems (narrow field of view, daytime, etc...), we have proposed using a catadioptric infrared camera. It permits to observe the scene independently of the luminosity conditions and acquire a large amount of information thanks to the wide field of view provided by the catadioptric mirror. For example, it is possible to observe a much larger part of the skyline, in any luminosity situations, which could improve the registration with a DEM. In this paper, we have extended our previous method of skyline detection towards catadioptric infrared images and experimental results have demonstrated the validity of our approach. The major contribution of this paper is the redefinition of the dynamic programming technique for catadioptric images using an adapted neighborhood and an appropriate θ forward direction. In this paper, dynamic programming was used for skyline detection and it could be applied for other purposes like image restoration [30], shape retrieval [31] or image matching [32] for example. Since catadioptric IR-reflective mirrors are not commercially available, we have also presented an interesting method to generate a catadioptric IR image from a real perspective IR image. The proposed method is easy to develop and by construction, it verifies the physical properties of an IR camera and the geometrical properties of a catadioptric camera. This is very useful for fast prototyping when cost and time constraints are important issues.

In future work, we plan to focus on sphere discretization, especially by applying different discretization steps in a dynamic manner. This would permit to focus attention in more interesting areas of the sphere/image and also obtain a faster algorithm.

VII. ACKNOWLEDGMENTS

This work has been performed within (1) the STAR project of the Hubert Curien (Egide) partnership between RCV Lab at KAIST-Korea and MIS at UPJV-France and (2) the project “Monocular vision based natural feature

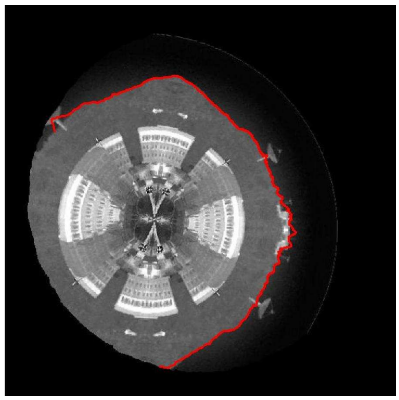


Fig. 9. Skyline extraction when the skyline is not entirely visible due to high camera rotation.

extraction for cognitive model" funded by Korea Ministry of Knowledge Economy.

REFERENCES

[1] T. Kanade, R. Collins, A. Lipton, P. Burt, and L. Wixson, "Advances in cooperative multi-sensor video surveillance," in *Darpa Image Understanding Workshop*, November 1998, pp. 3–24.

[2] P. Naval, "Camera pose estimation by alignment from a single mountain image," in *International Symposium on Intelligent Robotic Systems*, 1998, pp. 157–163.

[3] S. Ettinger, M. Nechyba, P. Ifju, and M. Waszak, "Vision-guided flight stability and control for micro air vehicles," *Advanced Robotics*, vol. 17, no. 7, pp. 617–640, 2002.

[4] L. Wang, S. Hsieh, E. Hsueh, F. Hsaio, and K. Hunag, "Complete pose determination for low altitude unmanned aerial vehicle using stereo vision," in *International Conference on Intelligent Robots and Systems (IROS'05)*, 2005, pp. 316–321.

[5] N. Winters, J. Gaspar, G. Lacey, and J. Santos-Victor, "Omni-directional vision for robot navigation," in *ICCV Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras (OMNIVIS'00)*, 2000.

[6] P. Chang and M. Hebert, "Omni-directional visual servoing for human-robot interaction," in *International Conference on Intelligent Robots and Systems (IROS'98)*, vol. 3, 1998, pp. 1801–1807.

[7] Y. Yagi, "Omnidirectional sensing and its applications," *IEICE Transactions On Information And Systems*, vol. E82D, no. 3, pp. 568–579, 1999.

[8] S. Hrabar and G. S. Sukhatme, "Omnidirectional vision for an autonomous helicopter," in *IEEE International Conference on Robotics and Automation (ICRA'03)*, 2003, pp. 557–564.

[9] S. Hrabar and G. S. Sukhatme, "A comparison of two camera configurations for optic-flow based navigation of a uav through urban canyons," in *International Conference on Intelligent Robots and Systems (IROS'04)*, 2004.

[10] C. Demonceaux, P. Vasseur, and C. Pegard, "Robust attitude estimation with catadioptric vision," in *International Conference on Intelligent Robots and Systems (IROS'06)*, 2006.

[11] C. Demonceaux, P. Vasseur, and C. Pegard, "Omnidirectional vision on UAV for attitude computation," in *IEEE International Conference on Robotics and Automation (ICRA'06)*, 2006.

[12] C. Demonceaux, P. Vasseur, and C. Pegard, "UAV attitude computation by omnidirectional vision in urban environment," in *IEEE International Conference on Robotics and Automation (ICRA'07)*, 2007.

[13] J. C. Bazin, I. S. Kweon, C. Demonceaux, and P. Vasseur, "Uav attitude estimation by vanishing points in catadioptric images," in *IEEE International Conference on Robotics and Automation (ICRA'08)*, May 2008.

[14] J. C. Bazin, I. S. Kweon, C. Demonceaux, and P. Vasseur, "A robust top-down approach for rotation estimation and vanishing points extraction by catadioptric vision in urban environment," in *International Conference on Intelligent Robots and Systems (IROS'08)*, September 2008.

[15] J. C. Bazin, I. S. Kweon, C. Demonceaux, and P. Vasseur, "UAV attitude estimation by combining horizon-based and homography-based approaches for catadioptric image," in *6th IFAC-EURON Intelligent Autonomous Vehicles 2007 (IAV 07)*, 2007.

[16] J. C. Bazin, C. Demonceaux, P. Vasseur and I. S. Kweon, "Motion esimation by catadioptric vision," in *Computer Vision and Image Understanding (CVIU) under submission*, 2008.

[17] D. Lowe, "Distinctive image features from scale-invariant keypoints," in *International Journal of Computer Vision (IJCV'03)*, vol. 20, 2003, pp. 91–110.

[18] C. Harris and M. Stephens, "A combined corner and edge detection," in *The Fourth Alvey Vision Conference*, 1988, pp. 147–151.

[19] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[20] F. Cozman and E. Krotkov, "Automatic mountain detection and pose estimation for teleoperation of lunar rovers," in *International Conference on Robotics and Automation (ICRA'97)*, vol. 3, 1997, pp. 2452–2457.

[21] C. Geyer and K. Daniilidis, "Catadioptric projective geometry," *International Journal of Computer Vision (IJCV'01)*, vol. 45, no. 3, pp. 223–243, 2001.

[22] S. Baker and S. K. Nayar, "A theory of catadioptric image formation," in *International Conference on Computer Vision (ICCV'98)*, 1998, pp. 35–42.

[23] J. P. Barreto and H. Araujo, "Geometric properties of central catadioptric line images and their application in calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI'05)*, vol. 27, no. 8, pp. 1327–1333, 2005.

[24] N. K. Logothetis, J. Pauls, and T. Poggio, "Shape representation in the inferior temporal cortex of monkeys," *Current Biology*, vol. 5, no. 5, pp. 552–563, 1995.

[25] J. H. Woo, I. S. Kweon, G. S. Kim, and I. C. Kim, "Robust horizon and peak extraction for vision-based navigation," in *IAPR Machine Vision Applications (MVA'05)*, 2005, pp. 526–529.

[26] A. A. Amini, T. E. Weymouth, and R. C. Jain, "Using dynamic programming for solving variational problems in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI'90)*, vol. 12, no. 9, pp. 855–867, 1990.

[27] C. Demonceaux and P. Vasseur, "Markov random fields for catadioptric image processing," *Pattern Recognition Letters*, vol. 27, no. 16, pp. 1957–1967, 2006.

[28] C. Garnier, R. Collorec, J. Flifla, C. Mouclier, and F. Rousee, "Infrared sensor modeling for realistic thermal image synthesis," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'99)*, 1999, pp. 3513–3516.

[29] Z. Wang, Q. Peng, Y. Lu, , and Z. Jiang, "A global infrared image synthesis model for large-scale complex urban scene," *International Journal of Infrared Millimeter Waves*, vol. 22, no. 8, 2001.

[30] M. Gong, "Images restoration using an iterative dynamic programming approach," in *Conference on Computer and Robot Vision (CRV'07)*, 2007, pp. 395–402.

[31] E. Milios and E. G. M. Petrakis, "Shape retrieval based on dynamic programming," *IEEE Transactions on Image Processing*, vol. 9, pp. 141–146, 2000.

[32] G. M. Quenot, "Image matching using dynamic programming: Application to stereovision and image interpolation," *Image Communication*, pp. 265–270, 1996.