Monocular Visual-IMU Odometry Using Multi-channel Image Patch Exemplars

Xingshuai Dong, Bo He, Xinghui Dong, and Junyu Dong

Department of Electronic Engineering, Ocean University of China, Qingdao, China
Center for Imaging Sciences, The University of Manchester, Manchester, United Kingdom
Department of Computer Science, Ocean University of China, Qingdao, China

Abstract—In this paper, we propose three sets of multi-channel image patch features for monocular visual-IMU (Inertial Measurement Unit) odometry. The proposed feature sets extract image patch exemplars from multiple feature maps of an image. We also modify an existing visual-IMU odometry framework by using different salient point detectors and feature sets and replacing the inlier selection approach with a self-adaptive scheme. The modified framework is used to examine the proposed feature sets. In addition to the Root Mean Square Error (RMSE) metric, we use the Hausdorff distance to measure the inconsistency between the estimated and ground-truth trajectories. Compared to the point-wise comparison used by RMSE, the Hausdorff distance takes the shape inconsistency of two trajectories into account and is hence more perceptually consistent. Experimental results show that the multi-channel feature sets outperform, or perform comparably to, the single gray level channel feature sets examined in this study. Particularly, the multi-channel feature set that uses integral channels, i.e., ICIMGP (Integral Channel Image Patches), outperforms two state-of-the-art feature sets: SIFT (Scale Invariant Feature Transform) and SURF (Speed Up Robust Features). Besides, ICIMGP performs better than the two multi-channel feature sets that are designed based on derivative channels and gradient channels respectively. These promising results are attributed to the fact that the multi-channel features encode richer image characteristics than their single gray level channel counterparts.

Keywords—Visual-IMU odometry; Odometry; Navigation; Multi-channel image features; Local features

1 Introduction

Accurate and reliable ego-motion estimation in dynamic and unknown surroundings plays an important role in the autonomous robot navigation and localization tasks. In contrast to the steady and invariant indoor environment, it is more challenging for robots to estimate their current location in highly dynamic outdoor environments [23]. In the literature, the Inertial Measure Unit (IMU) is one of the most commonly used sensors for solving this issue.
because it provides real-time carrier posture, velocity and position without being interfered by weather or light. However, the IMU severely suffers the error accumulation problem and may drift over a long time.

On the other hand, cameras are low-cost and easy to set up. The images acquired using cameras contain various surrounding information. The motion trajectory of cameras is normally estimated by tracking the matched (feature) points between consecutive image frames. Normally, cameras are able to successfully track features at low velocities. However, with the increase of velocity the accuracy rating of tracking is decreased due to the influence of motion blur and camera sampling rate [7]. Although the IMU has large measurement uncertainty at low-speed motion, it is able to capture high velocities and accelerations promptly. This characteristic makes cameras be complementary to the IMU system [7]. To be specific, the drift generated by the IMU can be reduced by combining the visual measurement with the IMU system. Therefore, the vison-aided inertial navigation system is able to achieve better performance than the pure IMU system.

Considering the characteristics of vision-aided inertial navigation systems, image features are key to these systems. When a gray level image is used, features are normally extracted from a single channel, e.g., the image itself. In contrast, multiple feature channels provide richer information. In this paper, we therefore adapt three sets of multi-channel image features. These feature sets encode different types of image characteristics. The proposed feature sets are tested using an established monocular visual-IMU odometry system [20] together with several state-of-the-art baseline feature sets. Since this system may encounter crashing when no inlier points are obtained, we use a simple self-adaptive inlier selection scheme instead of the original scheme. We used an extensively applied performance measure: the Root Mean Square Error (RMSE) metric in this study, which considers the accumulated error between the estimated and ground-truth trajectories based on the point-wise comparison. However, this metric does not take into account the shape inconsistency of trajectories. Thus, we use a second performance metric: the Hausdorff distance [12]. Since this metric considers the shape inconsistency of two trajectories, it is more perceptually consistent than the RMSE metric.

The contributions of this paper are: (1) we propose three sets of multi-channel image features for the visual-IMU odometry application; (2) we replace the inlier selection method of the odometry framework [20] using a self-adaptive scheme, which prevents the system from exceptionally crashing when an empty inlier set is selected; and (3) we use a shape matching algorithm [12] to measure the inconsistency between the estimated and ground-truth trajectories. To our knowledge, these aspects have not been addressed for monocular visual-IMU odometry systems in the community.

The rest of this paper is organized as follows. Related work is reviewed in Section 2. In Section 3, the proposed feature sets are introduced. The RMSE metric and the Hausdorff distance are described and compared in Section 4. The experimental setup is introduced in Section 5 and the results are reported in Section 6. Finally, our conclusions are drawn in Section 7.
2 Related Work

In this section, we review the previous work involved in visual(-IMU) odometry, the salient point detectors and local features used in visual odometry systems and multi-channel image features.

2.1 Visual (-IMU) Odometry

Motion trajectory estimation is a well-studied topic in the fields of computer vision and robotics. As an extensively applied technique to motion trajectory estimation, Visual Odometry (VO) incrementally estimates the motion trajectory of the camera using consecutive image frames [30]. According to the camera involved, VO can be divided into two categories: monocular and stereo [30]. In this study, we focus on the monocular VO application because of its simplicity and effectiveness.

Since IMU (Inertial Measurement Unit) and cameras are complementary, the combination of these improves both the reliability and precision of the motion trajectory estimation [22] [25]. In the literature, the tightly-coupled fusion solutions have received much attention. Inspired by the study of Davison et al. [8], Pinies et al. [27] proposed an Extended Kalman Filter (EKF) based real-time fusion framework using monocular vision. Instead of using the constant velocity motion model [8], the IMU motion model was utilized in this framework. On the basis of the work that Mourikis and Roumeliotis proposed [25], Hu and Chen [20] further introduced a sliding-window monocular visual-IMU odometry framework and achieved a tradeoff between the computational cost and precision. In addition, Shen et al. [31] presented a monocular vision based visual-inertial fusion framework in order to estimate the flight trajectory of the micro aerial vehicles (MAVs).

In contrast, the loosely-coupled fusion solutions consider IMU and the visual module as two separate parts. For instance, Sirtkaya et al. [32] fused the inertial navigation method with the relative pose estimation between consecutive image frames. However, regardless whether the tightly-coupled system or the loosely-coupled system is used, robust image feature sets are required in order to represent image sufficiently.

2.2 Salient Point Detectors and Local Features Used in Visual Odometry Systems

In Visual Odometry systems, the motion trajectory of vehicles is normally estimated by matching the consecutive frames recorded by cameras [30]. In order to enhance the image matching speed, a set of salient points are usually selected from images. Hence, salient point detection and feature extraction are two key components of those systems. In the applications of the vision based motion trajectory estimation, the most commonly used salient point detectors include the Harris [18] [22] and FAST corner detectors [29] [3], which are efficient but less distinctive and redetected.

In the computer vision community, two blob detectors: Difference of Gaussians (DoG) [24] and Fast Hessian [2] were proposed along with two local feature sets: Scale Invariant Feature Transform (SIFT) and Speed Up Robust Features (SURF), respectively. Compared with the corner detectors and associated image features, the SIFT and SURF feature sets are rotation-invariant. Both the feature sets have been used in visual (IMU) odometry systems.
In order to obtain both blob and corner points, Geiger et al. [16] introduced a blob and corner detector and a set of Sobel [33] filtered image patches based features. In addition, gray level image patch features have been used in visual (-IMU) odometry applications [3] [6] [26]. However, the local feature sets mentioned above are normally extracted from the single gray level channel of images. In contrast, multi-channel image representations provide richer image characteristics and own stronger discriminatory power.

2.3 Multi-channel Image Features

Multi-channel image features normally exploit different image feature channels and can be regarded as the extension of gray level features. Barros et al. [1] used the gray level image and the corresponding Sobel [33] filtered images computed at the horizontal and vertical directions for hand posture recognition. Considering the gradient data is associated with the image structure, Dong et al. [11] learned textons from the joint distributions of the gradient magnitude and gradient direction data computed using the Canny detector [4].

Recently, RGB-D images have been applied in computer vision applications. Dollár et al. [10] computed the depth and normal gradient data from RGB-D images in order to improve the performance of edge detection. Furthermore, Gupta et al. [17] exploited the combination of the height above ground, the angle with gravity and the horizontal disparity from RGB-D images. In addition, Dollár et al. [9] proposed a set of integral channel features. The integral channels are obtained from the linear and non-linear transformations of the original image. Those features have been successfully applied to object detection and contour detection [9].

Inspired by the studies mentioned above, we intend to investigate the application of multi-channel features to monocular visual-IMU odometry systems in this study. Our hypothesis is that multi-channel features outperform their counterparts computed from the gray level channel. Due to the efficiency and effectiveness of image patch features [34], we first extract local image patch exemplars from each channel separately and then concatenate these into a single feature vector.

3 Multi-channel Image Patch Features

It has been shown that multi-channel image features provide richer information than gray level features [1, 9, 10, 11, 17]. In this section, we introduce three sets of multi-channel image patch features, which will be used in the odometry framework described in Section 5. We first describe three different types of multi-channel data and then introduce the three multi-channel image patch feature sets.

Fig. 1 A gray level image frame contained in the KITTI dataset [15].
3.1 Multiple Derivative Channels

Image derivative maps can be computed using a variety of convolution filters, such as the Canny [4] and Sobel [33] operators. Considering the tradeoff between accuracy and the computational cost, we use the Canny operator [4] in this study. Given an input gray level image $I(x, y)$ (see Fig. 1 for example), the derivative maps $I_x(x, y)$ and $I_y(x, y)$ in the directions of $x$ and $y$ can be computed as:

\[
I_x(x, y) = \frac{\partial}{\partial x} (I \ast G) = I \ast \frac{\partial}{\partial x} G = I \ast G_x, \quad \text{(1)}
\]

\[
I_y(x, y) = \frac{\partial}{\partial y} (I \ast G) = I \ast \frac{\partial}{\partial y} G = I \ast G_y, \quad \text{(2)}
\]

where $G(x, y)$ is a Gaussian function, $G_x(x, y)$ and $G_y(x, y)$ are the derivative functions of $G(x, y)$ in the $x$ and $y$ directions respectively. The derivative maps obtained from the image shown in Fig. 1 are displayed in Fig. 2.

![Fig. 2 Two derivative maps: (a) the derivative map computed in the $x$ direction and (b) the derivative map calculated in the $y$ direction.](image)

3.2 Multiple Gradient Channels

Both image gradient magnitudes and gradient orientations can be used for image representation because they encode different structural characteristics of images. In essence, gradient magnitudes denote how quickly an image is changing, while gradient orientations mean in which direction the image color or intensity changes fast [11]. In addition, due to the influence of lighting conditions or camera properties, two images of the same scene may show different appearance. In this situation, the extracted features may fail to match the corresponding images. However, the gradient information has the merit of being less sensitive to lighting and camera changes. Therefore, the image features which are extracted from the gradient data computed from the original images are useful and robust.

In this study, the gradient maps are computed using the Canny operator [4]. Given the derivative maps: $I_x$ and $I_y$ computed using Equations (1) and (2) respectively, the gradient magnitude map can be calculated using the following equation:

\[
M(x, y) = \sqrt{I_x(x, y)^2 + I_y(x, y)^2}. \quad \text{(3)}
\]

Correspondingly, the gradient orientation map is computed using the equation below:

\[
O(x, y) = \tan^{-1}\left[\frac{I_y(x, y)}{I_x(x, y)}\right]. \quad \text{(4)}
\]

In terms of the image shown in Fig. 1, the gradient magnitude and orientation maps computed are shown in Fig. 3.
Fig. 3 Gradient channel maps: (a) the gradient magnitude map and (b) the gradient orientation map.

3.3 Integral Channels

Dollár et al. [9] introduced a set of integral channel features based on the linear and non-linear transformations of images. Compared with the traditional gray level or color features, the integral channel features provide more diverse but heterogeneous information. The integral channel method contains the normal channel(s), the gradient magnitude channel and six gradient histogram channels without smoothing. If the input image $I(x, y)$ is a gray level image, the normal channel $C(x, y)$ is the same as this image. In contrast, when a color image is considered, each color channel is treated as an individual normal channel. In addition to the normal channel(s), the other channels are computed from the linear or non-linear transformation of $I(x, y)$ [9]. The eight integral channel maps of the gray level image shown in Fig. 1 are displayed in Fig. 4.

3.4 Fast Hessian Salient Point Detector

In this study, we used the Fast Hessian salient point detector [2] to find a set of salient points from each gray level image frame. This detector was introduced as a part of the Speed Up Robust Features (SURF) [2] method. The detector was designed based on the determinant of the Hessian matrix. The Hessian matrix at the scale of $\sigma$ is expressed as [2]:

$$H(x, y, \sigma) = \begin{bmatrix}
\frac{\partial^2}{\partial x^2} G(\sigma) \ast I(x, y) & \frac{\partial}{\partial x} \frac{\partial}{\partial y} G(\sigma) \ast I(x, y) \\
\frac{\partial}{\partial x} \frac{\partial}{\partial y} G(\sigma) \ast I(x, y) & \frac{\partial^2}{\partial y^2} G(\sigma) \ast I(x, y)
\end{bmatrix}, \quad (5)$$

where $\frac{\partial}{\partial x} G(\sigma)$ and $\frac{\partial^2}{\partial x^2} G(\sigma)$ are the first order and second order Gaussian derivative functions respectively, $I(x, y)$ is the input image, and $\ast$ is the convolution function.

Due to the heavy computational cost of the high scale Gaussian convolution, Bay et al. [2] approximated the Laplacian of Gaussian functions using box filters. As a result, the speed of the Fast Hessian detector is dramatically increased. An accurate approximation of the Hessian determinant based on the approximated Gaussian functions is given as:

$$\text{det} (H_{\text{approx}}) = D_{xx}D_{yy} - (0.9D_{xy})^2, \quad (6)$$

where $D_{xx}$, $D_{yy}$, and $D_{xy}$ are the box filter approximations in the directions of $x$, $y$ and both $x$ and $y$ respectively. The detection of the local maxima of Equation (6) over different orientations and scales generates the salient
points of the input image. In our experiments, the Fast Hessian detector was applied to each gray level image frame and a set of salient points were obtained.

Fig. 4 Eight integral channel maps, including (a) the gray level channel, (b) the gradient magnitude channel and (c-h) six gradient histogram channels.

3.5 Multi-channel Image Patch Feature Sets

The straightforward representation of a pixel can be obtained as the image patch surrounding it. Compared to other image features, e.g., SIFT [24] and SURF [2], image patch based features retain original image characteristics while being more computationally efficient. Since local image patches often experience less distortion than global images, the similarity measurement between two local patches is more robust. In the literature, it has been shown that even small image patches can provide strong discriminatory power for texture images [34].

Considering the merits of multi-channel representations, we therefore use the image patches extracted from a series of different feature channels of an image to build a feature vector. The features are only extracted at the locations of the salient points detected using Fast Hessian [2] or other salient point detectors. Sample image patches extracted from the three types of multi-channel data described in the previous subsections are illustrated in Figures 5, 6 and 7 respectively. In terms of each salient point, the image patches extracted at this point in different
channels are first $L_2$ normalized and then concatenated into an individual feature vector. It should be noted that the gray level image is incorporated into the multiple derivative channels and multiple gradient channels data. (The gray level channel has been included in the integral channels representation in its original definition). The image patch feature sets extracted from multiple derivative channels (including the gray level channel), multiple gradient channels (including the gray level channel) and integral channels are named “DERP+IMGP”, “GRADP+IMGP” and “ICIMGP” respectively. The pipelines of these feature sets are illustrated in Figures 8, 9 and 10 respectively. We will test the three feature sets together with several state-of-the-art feature sets using a modified monocular visual-IMU odometry framework [20] in Section 6.

![Fig. 5](image5.png)  
**Fig. 5** Image patch exemplars (11×11) extracted from the two derivative channel maps which are computed at: (a) the $x$ direction and (b) the $y$ direction.

![Fig. 6](image6.png)  
**Fig. 6** Image patch exemplars (11×11) extracted from two gradient channel maps: (a) the gradient magnitude map and (b) the gradient orientation map.

![Fig. 7](image7.png)  
**Fig. 7** Image patch exemplars (11×11) extracted from the eight integral channel maps which are shown in Fig. 4.
4 Performance Measures

The Root Mean Square Error (RMSE) metric is normally used as performance measure for visual odometry systems. However, this measure only considers the accumulation of the error between the ground-truth position and the estimated position without taking into account the dissimilarity of the shape of trajectories. Therefore, in
addition to RMSE, we applied a shape matching algorithm, namely, the Hausdorff distance [12], to measure the inconsistency between the estimated and ground-truth trajectories. In essence, the Hausdorff distance compares two trajectories based on both local and global similarity and is more perceptually-consistent than RMSE.

4.1 Root Mean Squared Error (RMSE)

The RMSE measure is computed as the square root of the mean of the squared differences between the estimate data and the ground-truth data. When two sets of 2D coordinate data: A and B are used, the computation of RMSE is defined as:

\[
e(A,B) = \sqrt{\frac{\sum_{i=1}^{n}((x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2)}{n}}
\]

where \((x_i, y_i)\) belongs to the ground truth dataset A while \((\hat{x}_i, \hat{y}_i)\) is a member of the estimated dataset B.

As shown in Equation (7), RMSE only takes the accumulation of point-wise errors into account while ignoring the global dissimilarity of two trajectories. According to the definition, the RMSE measure is symmetric (i.e., \(e(A,B) = e(B,A)\)) because it computes the error between two points with the same index on two different trajectories. Fig. 11 illustrates the point-to-point comparison procedure used by RMSE. It can be seen that the matching from trajectory A to B is identical with that from trajectory B to A.
Fig. 11 Illustration of the point-wise matching used by RMSE: (a) the matching from trajectory $A$ to $B$ and (b) the matching from trajectory $B$ to $A$.

4.2 The Hausdorff Distance

In mathematics, the Hausdorff distance [12] is calculated as the maximal of the distances between a point in one set and its nearest point in another set. It measures the distance between two subsets of a metric space. The Hausdorff distance has been used for shape matching in computer vision. In this study, we used the modified Hausdorff distance introduced in [12]. Given two point sets: $A$ and $B$, the modified Hausdorff distance between these sets is computed as:

$$h(A,B) = \frac{1}{N_A} \sum_{a \in A} \{ \min_{b \in B} \{ d(a,b) \} \},$$

(8)

where $a$ and $b$ are the points in $A$ and $B$ respectively, and $d(a,b)$ is the direct distance between $a$ and $b$ (which is usually computed using the Euclidian distance). It can be learned from Equation (8) that the Hausdorff distance is asymmetric, i.e., $h(A,B) \neq h(B,A)$. Fig. 12 shows the difference between the matching computations from trajectory $A$ to $B$ and from trajectory $B$ to $A$. Therefore, a more general definition of the Hausdorff distance is expressed as:

$$H(A,B) = \max\{h(A,B), h(B,A)\}.$$  

(9)

Fig. 12 Illustration of the Hausdorff distance computation: (a) the computation from trajectory $A$ to $B$ and (b) the computation from trajectory $B$ to $A$.

4.3 Comparison of RMSE and the Hausdorff Distance

Three different paths are shown in Fig. 13, including the ground-truth path, Path 1 and Path 2. It can be observed that Path 2 is more similar to the ground-truth path than Path 1. However, the RMSE value 0.118 computed between the ground-truth path and Path 1 is the same as that calculated between the ground-truth path
and Path 2. In contrast, the Hausdorff distance between Path 1 and the ground-truth path is 0.091 while that computed between Path 2 and the ground-truth path is 0.063. This result suggests that the Hausdorff distance is more consistent with human perceptual judgements than the RMSE metric. Therefore, we used both RMSE and the Hausdorff distance as performance measures in this study.

![Graph showing ground-truth path and two estimations](image)

Fig. 13 The ground-truth path and its two different estimations.

5 Experimental Setup

Hu and Chen [20] proposed a visual-IMU odometry system which combines a monocular camera with an IMU. In this study, we modified this system in two aspects. First, we revised the salient point detection and feature extraction module in order that different salient point detectors and/or feature sets can be used in this system. Second, we replaced the feature matching and inlier selection module using a self-adaptive scheme in order to prevent the system from exceptionally crashing when insufficient inliers were returned. In addition, three paths selected from a publicly available dataset [15] were used along with the modified system.

5.1 The Modified Visual-IMU Odometry System

Inspired by the multi-state constraint Kalman filter [25], Hu and Chen [20] proposed a monocular visual-IMU odometry system (see Fig. 14 for pipeline) in which the trifocal tensor geometry relationship [19] between three images is used as camera measurement. This design releases the requirement to estimate the 3D position of feature points. The use of a moving-window scheme further accelerates the computational speed while retaining proper accuracy. The odometry system also applies the Random Sample Consensus (RANSAC) [13] method in order to reject the mismatched feature points or the feature points which locate on individual moving objects.
The pipeline of the visual-IMU odometry system proposed in [20].

5.1.1 Filter State Initialization

The filter state vector consists of the IMU state and the last two poses of the camera. The position, orientation, velocity, the IMU inner accelerator bias and the IMU inner gyroscope bias are comprised of the IMU state vector. The filter state includes two states: nominal and error. In this study, we used the same parameters as those used by Hu and Chen [20].

5.1.2 Filter Propagation

In the filter propagation step, prediction of the filter nominal state is performed using the IMU nominal state based on 4th-order Runge Kutta [5]. Prediction of the error state can be described as:

$$\hat{x} = F_c \hat{x}_k + G_c n_{IMU},$$  \hspace{1cm} (10)

where $F_c$ is the continuous-time state transition matrix, $G_c$ is the Jacobian matrix of the body error state, and $n_{IMU}$ is the system noise. The discrete time state transition matrix $F_d$ can be obtained by substituting $F_c$ into Taylor series:

$$F_d = I + F_c \Delta t + \frac{1}{2!} F_c^2 \Delta t^2 + \cdots.$$  \hspace{1cm} (11)

The continuous time system noise covariance matrix $Q_c$ is defined as:

$$Q_c = n_{IMU} n_{IMU}^T.$$  \hspace{1cm} (12)
while its discrete form $Q_d$ can be obtained as:

$$Q_d = \int_{t-1}^{T} F_d(t)G_dG_d^TF_d(t)^T\,dt.$$  

(13)

On the basis of $F_d$ and $Q_d$, the predicted error state covariance matrix $P_{k|k-1}$ can be calculated as:

$$P_{k|k-1} = F_dP_{k-1|k-1}F_d^T + Q_d.$$  

(14)

5.1.3 **Salient Point Detection and Feature Extraction**

When a new image is recorded by the camera, the system will conduct salient point detection and feature extraction. The reason for using a set of salient points rather than all the pixels in the image is to reduce the computational complexity. In this study, salient points were detected from gray level images. After salient point detection is complete, image features are extracted at the salient points detected. We modified the original framework [20] in order to incorporate different salient point detectors and image feature sets into it.

5.1.4 **Self-Adaptive Feature Matching and Inlier Selection**

The trifocal geometry relationship [19] between three consecutive image frames is used as the camera measurement data. In order to map the matched feature points between the first two frames into the third frame, the trifocal tensor [19] is utilized. In this study, we used the feature matching algorithm that Lowe [24] introduced. A “bucketing” method [21] is further used to choose a certain number of matched feature points for the purpose of minimizing the computational cost.

Since the dataset [15] used in this study was collected in the outdoor environment and contains various moving objects, these objects may lead to the mismatch of the points between consecutive frames or the match between independent objects. Hu and Chen [20] used the RANSAC method [13] to select inliers in order to address this problem. However, this method may return an insufficient or even empty inlier set, especially, when the feature matching operation is conducted strictly. Thus, we introduce a simple self-adaptive feature matching and inlier selection scheme. Specifically, the system will automatically adjust the threshold used by the feature matching algorithm and re-conduct the feature matching operation in the case that the inlier selection based on RANSAC fails. This process will be repeated until a proper number of inliers are returned. The self-adaptive scheme guarantees that the odometry system will not exceptionally crash when inlier selection fails.

5.1.5 **Filter State Update**

Once inliers are obtained, they are used to update the error state and error covariance of the filter based on the epipolar geometry model and the trifocal tensor model [19]. The nominal state of the filter is also corrected. Since only three poses are required for the filter state vector, the old state is replaced using the current state with the error covariance modified.

5.2 **Dataset**

Along with the odometry system described above, we used three paths (Path 1, Path 2 and Path 3) that Hu and Chen [20] selected from a publicly available and challenging dataset: the KITTI dataset [15]. Sample images of
these paths are illustrated in Fig. 15. The data was captured in the residential area or on the highway using a recording platform. This platform was equipped with multiple sensors, including two high-resolution stereo cameras (gray level and color), a Velodyne laser scanner, and an OXTS RT 3003 GPS/IMU localization unit with RTK correction signals. All sensors contained in this data recording system had been calibrated and synchronized. The resolution of the images used in this paper is 1241×376 pixels (Paths 1 and 2) or 1226×370 pixels (Path 3).

![Sample images of three experimental paths selected from the KITTI dataset](image)

**Fig. 15** Sample images of three experimental paths selected from the KITTI dataset [15].

5.3 Experimental Design

In our study, the modified visual-IMU odometry system was utilized in four experiments. The three paths contained in the KITTI dataset [15] were used in each experiment and only the synchronized gray level images were utilized. Regarding each path, the GPS/IMU localization unit data was used as ground-truth while the pure IMU method was used as a baseline. Both RMSE and the Hausdorff distance [12] were used as position error measures. In the first experiment, we tested three proposed multi-channel feature sets: DERP+IMGP, GRADP+IMGP and ICIMGP together with the Fast Hessian salient point detector [2]. In addition to the IMU data, three state-of-the-art feature sets: image patches (IMGP) [34], SIFT [24] and SURF [2] were used as baselines. In the second experiment, we investigated the effect of the size of image patches on the best multi-channel feature set found in the first experiment. We further examined the effect of salient point detectors on the best multi-channel feature set in the third experiment. Finally, in the fourth experiment we applied Principal Component Analysis (PCA) [35] to the best multi-channel feature set in order to reduce the dimensionality of feature vectors, and tested the compressed feature set together with the Fast Hessian detector [2].
6 Experimental Results

In this section, we report the results obtained in the four experiments that we introduced in Section 5.3.

6.1 Comparison of Different Multi-channel Feature Sets

We tested the three proposed multi-channel feature sets for the monocular visual-IMU odometry application in order to find the best feature set of these. The size of image patches was set as 11×11 pixels. Salient points were derived using the Fast Hessian salient point detector [2]. As shown in Fig. 16, in terms of each path, the ground-truth trajectory and the estimated trajectories obtained using different methods are plotted. In addition, the RMSE measure and the Hausdorff distance values computed between the ground truth trajectory and each estimated trajectory are reported in Table 1.

Table 1 The position RMSE and Hausdorff distance values computed between the ground-truth trajectory and the estimated trajectories obtained using IMU and different feature sets on (a) Path 1, (b) Path 2 and (c) Path 3.

| Algorithm       | Position RMSE (m) | Hausdorff Distance (m) |
|-----------------|-------------------|------------------------|
| IMU             | 2743.6794         | 1995.7541              |
| VIO ICIMGPG     | 4.3857            | 2.5333                 |
| VIO DERPG+IMGP  | 7.5590            | 4.0784                 |
| VIO GRADPG+IMGP | 11.8110           | 6.7101                 |
| VIO IMGP        | 20.9922           | 8.6838                 |
| VIO SIFT        | 8.7533            | 4.5310                 |
| VIO SURF        | 11.6636           | 6.1324                 |
| VO [16]         | 33.9685 [20]      | N/A                    |

(a)

| Algorithm       | Position RMSE (m) | Hausdorff Distance (m) |
|-----------------|-------------------|------------------------|
| IMU             | 64.3528           | 19.1202                |
| VIO ICIMGPG     | 41.7974           | 11.5592                |
| VIO DERPG+IMGP  | 67.2520           | 27.7012                |
| VIO GRADPG+IMGP | 99.8435           | 39.3966                |
| VIO IMGP        | 140.7800          | 80.4604                |
| VIO SIFT        | 87.9880           | 32.5323                |
| VIO SURF        | 83.9164           | 43.3295                |
| VO [16]         | 596.3744 [20]     | N/A                    |

(b)

| Algorithm       | Position RMSE (m) | Hausdorff Distance (m) |
|-----------------|-------------------|------------------------|
| IMU             | 6741.0163         | 4641.1988              |
| VIO ICIMGPG     | 23.0707           | 11.0590                |
| VIO DERPG+IMGP  | 55.5414           | 29.4891                |
| VIO GRADPG+IMGP | 38.0552           | 19.4898                |
| VIO IMGP        | 70.7277           | 38.6420                |
| VIO SIFT        | 40.1095           | 19.4118                |
| VIO SURF        | 23.1811           | 12.1333                |
| VO [16]         | 211.2474 [20]     | N/A                    |

(c)
Fig. 16 The ground-truth trajectory and the trajectories obtained using IMU and different feature sets on: (a) Path 1 (residential area, \(\approx 540\text{m}, \approx 25\text{km/h}\)); (b) Path 2 (highway, \(\approx 2200\text{m}, \approx 82\text{km/h}\)); and (c) Path 3 (residential area, \(\approx 3580\text{m}, \approx 29\text{km/h}\)) (Map source: GoogleMap).
It can be seen that: (1) IMU often suffers the error accumulation issue when the path is long and complicated (see Figs. 16 (a) and (c)). However, it performs properly when the path is straight and the speed is high (see Fig. 16 (b)). This should be attributed to the fact that IMU is suitable for the case of the high speed, linear motion; (2) the combination of monocular camera and IMU can reduce the drift issue; (3) the three multi-channel feature sets perform better than the gray level image patch feature set, and outperform, or perform comparably to, SIFT [24] and SURF [2]; (4) the multi-channel feature set using integral channels, i.e., ICIMGP, outperforms all its counterparts tested in this experiment regardless of whether the experiment is conducted in the residential area or on the highway; (5) no matter which performance measure is used, the performance of all the methods obtained on different paths are similar.

In addition, we compare the results obtained in this experiment with those derived using the pure monocular visual odometry method [16] (referred to as “VO”) by Hu and Chen [20]. The position RMSE values obtained using this method on the three paths are 33.9685m, 596.3744m and 211.2474m respectively [20]. (Note that the Hausdorff distance data is not available in this case). As illustrated in Table 1, the position RMSE values derived using the VIO methods based on the three proposed multi-channel feature sets are much smaller than these values.

Since the multi-channel feature set based on integral channels performs better than the other multi-channel feature sets, we only examine it in the following experiments.

6.2 The Effect of the Size of Image Patches

The advantage of the proposed integral channel image patch (ICIMGP) feature set over its counterparts has been shown in the previous experiment. In this experiment, we investigated the effect of the size of image patches on the ICIMGP feature set. The size of image patches was set as 7×7, 9×9, 11×11, 13×13 and 15×15 pixels. The IMU data was used as a baseline.

The ground-truth trajectory and the trajectories obtained using IMU and the ICIMGP feature set with different sizes of image patches are shown in Fig. 17. In addition, the overall position RMSE and Hausdorff distance values are reported in Table 2. It can be observed that: (1) the ICIMGP feature set performs better, or comparably to, the IMU method when different sizes of image patches are used; (2) the ICIMGP feature set performs better when 11×11 image patches are used than when the other sizes of patches are utilized; and (3) no matter the experiment is performed in the residential area or on the highway, the ICIMGP feature set with different sizes of image patches performs properly. It should be noted that the performance of ICIMGP is not improved when the size of image patches is beyond 11×11 pixels. This finding is similar to that Gauglitz et al. [14] observed.
Fig. 17 The ground-truth trajectory and the trajectories obtained using IMU and the ICIMGP feature set with different sizes of image patches on: (a) Path 1 (residential area, \( \approx 540 \text{m}, \approx 23 \text{km/h} \)); (b) Path 2 (highway, \( \approx 2200 \text{m}, \approx 82 \text{km/h} \)); and (c) Path 3 (residential area, \( \approx 3580 \text{m}, \approx 29 \text{km/h} \)) (Map source: GoogleMap).
Table 2 The position RMSE and Hausdorff distance values computed between the ground-truth trajectory and the estimated trajectories derived using IMU and the ICIMGP feature set with different sizes of image patches on (a) Path 1, (b) Path 2 and (c) Path 3.

| Algorithm | Position RMSE (m) | Hausdorff Distance (m) |
|-----------|------------------|------------------------|
| IMU       | 2743.6794        | 1995.7541              |
| VIO N=7   | 12.6163          | 6.3257                 |
| VIO N=9   | 8.8274           | 4.2856                 |
| VIO N=11  | 4.3857           | 2.5333                 |
| VIO N=13  | 7.9171           | 4.2269                 |
| VIO N=15  | 6.9833           | 3.4828                 |

(a)

| Algorithm | Position RMSE (m) | Hausdorff Distance (m) |
|-----------|------------------|------------------------|
| IMU       | 64.3528          | 19.1202                |
| VIO N=7   | 59.1876          | 21.8493                |
| VIO N=9   | 46.6500          | **9.2421**             |
| VIO N=11  | **41.7974**      | 11.5592                |
| VIO N=13  | 53.4276          | 16.6151                |
| VIO N=15  | 59.8459          | 21.1565                |

(b)

| Algorithm | Position RMSE (m) | Hausdorff Distance (m) |
|-----------|------------------|------------------------|
| IMU       | 6741.0165        | 4641.1988              |
| VIO N=7   | 42.7734          | 21.1739                |
| VIO N=9   | 38.9364          | 18.9013                |
| VIO N=11  | **23.0707**      | **11.0590**            |
| VIO N=13  | 38.6323          | 19.9826                |
| VIO N=15  | 34.4089          | 19.2121                |

(c)

It should be noted that the position RMSE value obtained using the 9×9 ICIMGP feature set on Path 2 is greater than that obtained using the 11×11 ICIMGP feature set. However, it is not the case when the Hausdorff distance is considered. Fig. 18 shows the trajectories obtained using both the feature sets. It can be seen that the trajectory obtained using the 9×9 ICIMGP feature set is more similar with the ground-truth one than that derived using the 11×11 ICIMGP feature set. Here, the advantage of the Hausdorff distance over the RMSE metric is shown.

Fig. 18 The ground-truth trajectory and the trajectories obtained using the ICIMGP feature sets when 9×9 and 11×11 image patches are used (Map source: GoogleMap).
6.3 The Effect of Salient Point Detectors

In this experiment, we tested the ICIMGP feature set along with different salient point detectors. In addition to the Fast Hessian detector [2], we examined the Blob and Corner detector [16] and the Difference of Gaussian (DoG) detector [24]. The 11×11 image patches were used for the ICIMGP feature set. Besides the IMU method, we compared the ICIMGP feature set with two original combinations of salient point detector and local feature set: the DoG detector and SIFT (DoG+SIFT) [24] and the Fast Hessian detector and SURF (FASTHESSIAN+SURF) [2].

Fig. 19 shows the ground-truth trajectory and the estimated trajectories obtained using IMU and different combinations of detector and feature set. Table 3 further lists the overall RMSE and Hausdorff distance values computed between the estimated trajectories and the ground truth trajectory. It can be learned that the ICIMGP feature set performs better when combined with the Fast Hessian detector [2] than combined with the Blob and Corner [16] or Difference of Gaussian [24] detectors, in terms of both performance measures. As can be seen, the best combination also outperforms two state-of-the-art combinations: DoG+SIFT [24] and FASTHESSIAN+SURF [2], which have been extensively used in various computer vision applications.

Table 3 The position RMSE and Hausdorff distance values computed between the ground-truth trajectory and the estimated trajectories derived using IMU and the ICIMGP feature set with different salient point detectors on (a) Path 1, (b) Path 2 and (c) Path 3.

| Algorithm                  | Position RMSE (m) | Hausdorff Distance (m) |
|----------------------------|-------------------|------------------------|
| IMU                        | 2743.6794         | 1995.7541              |
| VIO BLOB&CORNER+ICIMGP     | 6.0011            | 3.1376                 |
| VIO DOG+ICIMGP             | 6.4074            | 3.5519                 |
| VIO FASTHESSIAN+ICIMGP     | 4.3857            | 2.5333                 |
| VIO DOG+SIFT               | 4.6816            | 2.5342                 |
| VIO FASTHESSIAN+SURF       | 11.6636           | 6.1324                 |

(a)

| Algorithm                  | Position RMSE (m) | Hausdorff Distance (m) |
|----------------------------|-------------------|------------------------|
| IMU                        | 64.3528           | 19.1202                |
| VIO BLOB&CORNER+ICIMGP     | 54.9509           | 16.3611                |
| VIO DOG+ICIMGP             | 107.1095          | 21.7947                |
| VIO FASTHESSIAN+ICIMGP     | 41.7974           | 11.5592                |
| VIO DOG+SIFT               | 96.6075           | 21.0811                |
| VIO FASTHESSIAN+SURF       | 83.9164           | 43.3295                |

(b)

| Algorithm                  | Position RMSE (m) | Hausdorff Distance (m) |
|----------------------------|-------------------|------------------------|
| IMU                        | 6741.0165         | 4641.1988              |
| VIO BLOB&CORNER+ICIMGP     | 27.4207           | 13.1486                |
| VIO DOG+ICIMGP             | 41.3388           | 20.4872                |
| VIO FASTHESSIAN+ICIMGP     | 23.0707           | 11.0590                |
| VIO DOG+SIFT               | 28.3642           | 14.3985                |
| VIO FASTHESSIAN+SURF       | 23.1811           | 12.1333                |

(c)
Fig. 19 The ground-truth trajectory and the trajectories obtained using IMU and different combinations of detector and feature set on: (a) Path 1 (residential area, $\approx 540m$, $\approx 25km/h$); (b) Path 2 (highway, $\approx 2200m$, $\approx 82km/h$); and (c) Path 3 (residential area, $\approx 3580m$, $\approx 29km/h$) (Map source: GoogleMap).
6.4 Principal Component Analysis (PCA) Based ICIMGP

It has been shown in the previous three experiments that the proposed ICIMGP feature set outperforms its counterparts when combined with the Fast Hessian [2] detector. However, the dimensionality of ICIMGP feature vectors is high, which results in the slow speed of feature matching. To solve this problem, we applied Principal Component Analysis (PCA) [35] to ICIMGP feature vectors in order to obtain more compact features. This compressed ICIMGP feature set is referred to as PCA-ICIMGP in this paper. In this experiment, we evaluated different dimensional PCA-ICIMGP feature sets together with the Fast Hessian detector. The 11×11 image patches were used and the dimensionality of this feature set was set as 64, 128, 256 and 512. The IMU method and the original ICIMGP feature set were used as baselines.

Fig. 20 displays the ground-truth trajectory and the estimated trajectories obtained using IMU, ICIMGP and different dimensional PCA-ICIMGP feature sets. Table 4 further reports the overall RMSE and Hausdorff distance values computed between the estimated trajectories and the ground-truth trajectory. It can be seen that the 256D PCA-ICIMGP feature set outperforms all its counterparts, including the original ICIMGP feature set, on the three paths. However, the feature matching speed is dramatically accelerated because the dimensionality of PCA-ICIMGP feature vectors is much lower than that of ICIMGP feature vectors (968D).

Table 4  The position RMSE and Hausdorff distance values computed between the ground-truth trajectory and the estimated trajectories derived using IMU, the ICIMGP feature set and different dimensional PCA-ICIMGP feature sets on (a) Path 1, (b) Path 2 and (c) Path 3.

| Algorithm       | Position RMSE (m) | Hausdorff Distance (m) |
|-----------------|-------------------|-------------------------|
| IMU             | 2743.6794         | 1995.7541               |
| VIO ICIMGP      | 4.3857            | 2.5333                  |
| VIO PCA-ICIMGP (64D) | 7.5973         | 3.3419                  |
| VIO PCA-ICIMGP (128D) | 5.4570        | 3.0101                  |
| VIO PCA-ICIMGP (256D) | 3.5758        | 2.2646                  |
| VIO PCA-ICIMGP (512D) | 5.7573         | 2.6985                  |

(a)

| Algorithm       | Position RMSE (m) | Hausdorff Distance (m) |
|-----------------|-------------------|-------------------------|
| IMU             | 64.3528           | 19.1202                 |
| VIO ICIMGP      | 41.7974           | 11.5592                 |
| VIO PCA-ICIMGP (64D) | 82.2189        | 19.8431                 |
| VIO PCA-ICIMGP (128D) | 71.1884        | 16.9349                 |
| VIO PCA-ICIMGP (256D) | 36.3533        | 10.1799                 |
| VIO PCA-ICIMGP (512D) | 47.1887         | 16.7509                 |

(b)

| Algorithm       | Position RMSE (m) | Hausdorff Distance (m) |
|-----------------|-------------------|-------------------------|
| IMU             | 6741.0165         | 4641.1988               |
| VIO ICIMGP      | 23.0707           | 11.0590                 |
| VIO PCA-ICIMGP (64D) | 35.2199        | 18.7608                 |
| VIO PCA-ICIMGP (128D) | 33.2687        | 17.3726                 |
| VIO PCA-ICIMGP (256D) | 22.5784        | 11.1139                 |
| VIO PCA-ICIMGP (512D) | 28.2273         | 13.8400                 |

(c)
Fig. 20 The ground-truth trajectory and the trajectories obtained using IMU, ICIMGP and different dimensional PCA-ICIMGP feature sets on: (a) Path 1 (residential area, \( \approx 540 \text{m}, \approx 25 \text{km/h} \)); (b) Path 2 (highway, \( \approx 2200 \text{m}, \approx 82 \text{km/h} \)); and (c) Path 3 (residential area, \( \approx 3580 \text{m}, \approx 29 \text{km/h} \)) (Map source: GoogleMap).
7 Conclusions

In this paper, we proposed three different sets of multi-channel image patch features for Monocular Visual-IMU Odometry. Compared to the single gray level channel image feature sets, the proposed feature sets are able to exploit richer image characteristics. We modified the monocular visual-IMU odometry system that Hu and Chen [20] proposed. Different salient point detectors or image feature sets can be incorporated into this system. A self-adaptive inlier selection scheme was also used to replace the original scheme. This new scheme prevents the system from exceptionally crashing due to the selection of insufficient inliers. In addition to the RMSE metric, we used the Hausdorff distance [12] to measure the inconsistency between the estimated and ground-truth trajectories. This distance measure compares two trajectories based on both local and global similarity and is more perceptually-consistent than RMSE.

The modified odometry system was applied to three different paths [15] in order to test the proposed multi-channel feature sets. The GPS/IMU localization unit data with regard to each path was used as ground-truth. The pure IMU method, the gray level image patch [34], SIFT [24] and SURF [2] feature sets were used as baselines. Experimental results show the advantages of the multi-channel image feature sets over single gray level channel feature sets. In particular, the multi-channel feature set based on the 11×11 integral channel [9] image patches outperformed all its counterparts tested in this study, including two state-of-the-art feature sets: SIFT [24] and SURF [2], when it was used together with the Fast Hessian salient point detector [2]. It is noteworthy that the performance was slightly enhanced but the feature matching speed was dramatically accelerated when PCA [35] was used to reduce the dimensionality of ICIMGP feature vectors. We attribute these promising results to the fact that the multi-channel feature sets encode more diverse image characteristics than their single gray level channel counterparts.

Acknowledgement

This work is partially supported by the Natural Science Foundation of China (41176076) and the High Technology Research and Development Program of China (2014AA093410).

References

[1] Barros P, Magg S, Weber C, Wermter S (2014) A multichannel convolutional neural network for hand posture recognition. In: Proceedings of International Conference on Artificial Neural Networks, pp 403-410
[2] Bay H, Ess A, Tuytelaars T, Van G L (2008) Speeded-up robust features (SURF). Computer Vision and Image Understanding 110(3):346-359
[3] Bloesch M, Omari S, Hutter M, Siegwart R (2015) Robust visual inertial odometry using a direct EKF-based approach. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 298-304
[4] Canny J (1986) A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 1986(6):679-698
[5] Cartwright J H E, Piro O (1992) The dynamics of Runge-Kutta methods. International Journal of Bifurcation and Chaos 1992 2(03):427-449
[6] Cheng Y, Maimone M W, Matthies L (2006) Visual odometry on the Mars exploration rovers-a tool to ensure accurate driving and science imaging. IEEE Robotics & Automation Magazine 13(2):54-62
[7] Corke P, Lobo J, Dias J (2007) An introduction to inertial and visual sensing. International Journal of Robotics Research 26(6):519-535
[8] Davison A J, Reid I D, Molton N D, Stasse O (2007) MonoSLAM: Real-time single camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(6):1052-1067
[9] Dollár P, Tu Z, Perona P, Belongie S (2009) Integral channel features. In: Proceedings of British Machine Vision Conference 2009
[10] Dollár P, Zitnick C L (2015) Fast edge detection using structured forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 37(8):1558-1570
[11] Dong X, Dong J, Wang S, Chantler M J (2015) Perceptual texture retrieval using spatial distributions of textons (SDoT). In: Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, pp 663-666
[12] Dubuisson M P, Jain A K (1994) A modified Hausdorff distance for object matching. In: Proceedings of 12th International Conference on Pattern Recognition, Vol. 1, pp 566-568
[13] Fischler M A, Bolles R C (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24(6):381-395
[14] Gauglitz S, Höllerer T, Turk M (2011) Evaluation of interest point detectors and feature descriptors for visual tracking. International Journal of Computer Vision 94(3):335-360
[15] Geiger A, Lenz P, Stiller C, Urtasun R (2013) Vision meets robotics: The KITTI dataset. International Journal of Robotics Research (2013)32:1229-1235
[16] Geiger A, Ziegler J, Stiller C (2011) StereoScan: Dense 3d reconstruction in real-time. In: Proceedings of IEEE Intelligent Vehicles Symposium, pp 963-968
[17] Gupta S, Girshick R, Arbeláez P, Malik J (2014) Learning rich features from RGB-D images for object detection and segmentation. In: Proceedings of IEEE International Conference on Computer Vision, pp 345-360
[18] Harris C, Stephens M (1988) A combined corner and edge detector. In: Proceedings of Alvey Vision Conference, pp 147-151
[19] Hartley R, Zisserman A (2003) Multiple view geometry in computer vision (2nd ed). Cambridge University Press, Cambridge
[20] Hu J S, Chen M Y (2014) A sliding-window visual-imu odometer based on tri-focal tensor geometry. In: Proceedings of IEEE International Conference on Robotics and Automation, pp 3963-3968
[21] Kitt B, Geiger A, Lategahn H (2010) Visual odometry based on stereo image sequences with RANSAC-based outlier rejection scheme. In: Proceedings of IEEE Intelligent Vehicles Symposium, pp 486-492
[22] Li M, Mourikis A I (2012) Improving the accuracy of EKF-based visual-inertial odometry. In: Proceedings of IEEE International Conference on Robotics and Automation, pp 828-835
[23] Liu B, Adams M, Ibáñez-Guzmán J (2005) Multi-aided inertial navigation for ground vehicles in outdoor uneven environments. In: Proceedings of IEEE International Conference on Robotics and Automation, pp 4703-4708
[24] Lowe D G (2004) Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2):91-110
[25] Mourikis A I, Roumeliotis S I (2007) A multi-state constraint Kalman filter for vision-aided inertial navigation. In: Proceedings of IEEE International Conference on Robotics and Automation, pp 3565-3572
[26] Nistér D, Naroditsky O, Bergen J (2004) Visual odometry. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 652-659
[27] Pinhès P, Lapton T, Sukkarieh S, Tardós J D (2007) Inertial aiding of inverse depth SLAM using a monocular camera. In: Proceedings of IEEE International Conference on Robotics and Automation, pp 2797-2802
[28] Rodriguez D, Aouf N (2012) Robust egomotion for large-scale trajectories. In: Proceedings of IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems, pp 156-161
[29] Rosten E, Drummond T (2006) Machine learning for high-speed corner detection. In: Proceedings of European Conference on Computer Vision, pp 430-443
[30] Scaramuzza D, Fraundorfer F (2011) Visual odometry [Tutorial]. IEEE Robotics & Automation Magazine, 18(4):80-92
[31] Shen S, Michael N, Kumar V (2015) Tightly-coupled monocular visual-inertial fusion for autonomous flight of rotorcraft MAVs. In: Proceedings of IEEE International Conference on Robotics and Automation, pp 5303-5310
[32] Sirtkaya S, Seymen B, Alatan A A (2013) Loosely coupled Kalman filtering for fusion of visual odometry and inertial navigation. In: Proceedings of International Conference on Information Fusion, pp 219-226
[33] Sobel I, Feldman G (1968) A 3x3 isotropic gradient operator for image processing. Presented at the Stanford Artificial Intelligence Project in 1968

[34] Varma M, Zisserman A (2009) A statistical approach to material classification using image patch exemplars. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(11):2032-2047

[35] Wold S, Esbensen K, Geladi P (1987) Principal component analysis. Chemometric and Intelligent Laboratory Systems 2(1-3): 37-52