



**TECHNISCHE  
UNIVERSITÄT  
DRESDEN**



---

**Fakultät Elektrotechnik und Informationstechnik** Institut für Automatisierungstechnik

---

# BACHELOR THESIS

zum Thema

Image Based Visual Servoing for Aerial Robot

vorgelegt von Pablo Rodríguez Robles  
im Studiengang Aerospace Engineering, Jg. 2014  
geboren am 28.02.1996 in León, Spain

Betreuer: Dipl.-Ing. Chao Yao  
Verantwortlicher Hochschullehrer: Prof. Dr. techn. Klaus Janschek  
Tag der Einreichung: 27.03.2018

# Aufgabenstellung

Test der PDF-Integration

# Achtung

Auch wenn die Möglichkeit besteht, die eingescannte Aufgabenstellung als PDF zu integrieren, muss in **einem einzureichendem Exemplar** die Aufgabenstellung **im Original** eingebunden werden.



**TECHNISCHE  
UNIVERSITÄT  
DRESDEN**



---

**Fakultät Elektrotechnik und Informationstechnik** Institut für Automatisierungstechnik

---

## **Image Based Visual Servoing for Aerial Robot**

Hier muss der Text für die deutsche Kurzfassung inklusive eines aussagekräftigen Bildes eingefügt werden.

Betreuer:	Dipl.-Ing. Chao Yao
Hochschullehrer:	Prof. Dr. techn. Klaus Janschek
Tag der Einreichung:	27.03.2018

---

Bearbeiter: Pablo Rodríguez Robles



**TECHNISCHE  
UNIVERSITÄT  
DRESDEN**



---

**Fakultät Elektrotechnik und Informationstechnik** Institut für Automatisierungstechnik

---

## **Image Based Visual Servoing for Aerial Robot**

Here an English abstract including one significant image must be inserted.

Tutor: Dipl.-Ing. Chao Yao  
Supervisor: Prof. Dr. techn. Klaus Janschek  
Day of Submission: 27.03.2018

---

**STUDENT RESEARCH THESIS**

**Author: Pablo Rodríguez Robles**

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation and Background . . . . .	1
1.2	Aims and Objectives . . . . .	2
<b>2</b>	<b>Theoretical Background and State of the Art</b>	<b>3</b>
2.1	Visual Servoing Theoretical Basics . . . . .	3
2.2	State of the Art . . . . .	3
2.2.1	Visual Servoing for Aerial Robots . . . . .	3
2.2.2	Visual Servoing for Aerial Manipulators . . . . .	6
<b>3</b>	<b>Software Requirements Specification and Structured Analysis</b>	<b>11</b>
3.1	Software Requirement Specification . . . . .	11
3.1.1	Product Perspective . . . . .	11
3.1.2	User Characteristics . . . . .	12
3.1.3	Assumptions and Dependencies . . . . .	12
3.1.4	Functional Requirements . . . . .	12
3.1.5	Other Requirements . . . . .	13
3.1.6	General Constraints . . . . .	13
3.2	Structured Analysis . . . . .	13
<b>4</b>	<b>Visual Servoing Algorithm Description</b>	<b>14</b>
<b>5</b>	<b>Implementation of the Visual Servoing Controller</b>	<b>15</b>
<b>6</b>	<b>Final Results and Conclusions</b>	<b>16</b>
<b>7</b>	<b>Future Work</b>	<b>17</b>

## List of Figures

## List of Tables

2.1	Overview of the different approaches for VS in flying manipulators	10
-----	--	----

## List of Listings



# 1 Introduction

## 1.1 Motivation and Background

During the last decade, the use of Unmanned Aerial Vehicles (UAVs) has spread among very different applications. The use of flying robots can be very helpful to improve the way some tasks are already achieved by terrestrial robots. For example, object transportation, environment mapping or surveillance. At the Institute of Automation Engineering<sup>1</sup> of the Technical University of Dresden, a drone is being developed in cooperation with the Institute of Solid Mechanics<sup>2</sup> to investigate the use of flying robots in aerial manipulation.

When dealing with manipulation of objects, it is desired that the aerial robot adopts a certain pose with respect to the target before the manipulation process really starts. The present work deals with the development of a Visual Servoing (VS) control system that helps a quadrotor robot to acquire the desired pose by means of image data.

A monocular monochrome camera as well as an Inertial Measurement Unit (IMU) are planned to be the only available on board sensors. For the controller proposed the feedback is directly computed from image features rather than estimating the robot's pose and using the pose errors as control input.

In order to integrate the visual servoing algorithm into the future modular robot system, the algorithm has been designed and tested on a underactuated conventional quadrotor. The aerial robot is implemented within the ROS<sup>3</sup> framework, where the visual servoing controller developed for this thesis is also integrated. Instead of using real hardware the complete system is simulated using Gazebo<sup>4</sup>.

---

<sup>1</sup>Technische Universität Dresden. Institut für Automatisierungstechnik. 01062 Dresden, Germany

<sup>2</sup>Technische Universität Dresden. Institut für Festkörpermechanik. 01062 Dresden, Germany

<sup>3</sup>[www.ros.org](http://www.ros.org)

<sup>4</sup>[www.gazebo-sim.org](http://www.gazebo-sim.org)

## 1.2 Aims and Objectives

The aim of this work is to implement and test a VS control algorithm for a quadrotor, which could be later used by the Flypulator (TODO: Add reference) project. This includes the review of the state of the art with regard to Visual Servoing, the design of a solution and a prototypical implementation with in the ROS framework and simulation with Gazebo of a test case.

The present thesis documents comprehensively the theoretical background, implementation details and results of the conducted work through the following structure. In Chapter 2 the theoretical background and state of the art of Visual Servoing is presented. Chapter 3 gives a description of the system requirements as well as the system decomposition by Structure Analysis (SA) **SA\_Braune** Chapter 4 describes the solution developed and the algorithms to be tested. Chapter 5 deals with the implementation, testing and validation. Finally, Chapter 6 contains the final results and conclusions and Chapter 7 suggests future improvement and research paths.

## 2 Theoretical Background and State of the Art

### 2.1 Visual Servoing Theoretical Basics

### 2.2 State of the Art

#### 2.2.1 Visual Servoing for Aerial Robots

In this section, the literature on the use of Visual Servoing to control the translation of aerial robots is analyzed. The focus of this review is on the IBVS approach, where image feature errors between the current and target image are mapped to actuator inputs through the inverse of the Jacobian matrix. The different approaches presented complete the basic IBVS formulation with alternatives on feature selection, Jacobian matrix construction and different control strategies to make the error converge to zero, and with it, the relative pose between UAV and target object will converge to the desired one.

The task for the system in this case consists of positioning the aerial robot in a desired position with respect to a 2D target using only the translation and rotation of the quadrotor.

The Visual Servoing research was originally developed for serial manipulators. In order to translate these techniques to aerial robots there are different aspects that may be considered:

- Feature depth estimation.
- Dynamics of the system.
- Under-actuation of the system.

To cope with these problems different solutions have been proposed and applied for the usual configuration of a quadrotor helicopter, with the camera mounted on its platform (eye-in-hand configuration).

In the past, several approaches have been taken to face the problem of depth estimation, such as: partial pose estimation (see **malis\_2\_1999**), adaptive

control (see **Papanikolopoulos**) or estimation of the Jacobian using quasi-Newton methods (see **Piepmeyer**). Nowadays, the most frequent approach is the use of a partitioned control that allows the uncoupling of translational and rotational degrees of freedom<sup>1</sup>.

For serial manipulators, a low-level actuator is usually employed to compensate the dynamic behavior of the system, making possible to control it with velocity commands as a first order system. For a full-dynamics quadrotor model accounting for aggressive maneuvers this is not at all the case, since it is a fourth order system. Thus, other strategies are usually considered to control such a robot (see **Mellinger**). Already in the basic literature on Visual Servoing (see **chaumette\_visual\_2007**) the authors suggest that kinematic control may not be enough for the case of aggressive maneuvers.

When dealing with the dynamics it is necessary to incorporate the measurement of linear and angular velocities, usually coming from an IMU. Dynamic control leads to a coupling of the camera ego-motion on the image plane.

Image moments are invariant to some transformations like scale, 2D translation or 2D rotation. This has been intensively used in pattern recognition.

Introducing visual features based on image moments allows to design a decoupled control scheme when the object is parallel to the image plane (see **tahri\_2005**). Later, it is possible to generalize this property for the case of a non-parallel position with respect to the image plane.

One of the major questions in Visual Servoing is the selection of an adequate set of visual features and combine them in a proper way. It can be just a set of image points, lines, circles or rather a more complex set of visual moments. A proper combination should lead to

In the work **guenard\_2008** an IBVS controller for the full dynamics of a quadrotor system around the hovering position is proposed. The biggest difficulty to develop dynamic controllers based on IBVS is the high coupling in the image Jacobian between the rotational and translational dynamics. Something that does not happen in the case of PBVS. The paper introduces a new visual error formulation to improve the conditioning of the Jacobian matrix. Image moment features are augmented with the information provided

---

<sup>1</sup>However, it is important to bear in mind that due to the under-actuated dynamics of an usual 4 DOF quadrotor, it is only possible to control the three linear velocities and the yaw (rotational) speed. While the other two rotational speeds, roll and pitch, are used to move the thrust vectors of the propellers and thus generate movement on the horizontal plane.

by the IMU and used in conjunction with a non-linear controller.

Common strategy is to use two different loops to separate the control problem. The inner attitude loop runs at high gain using IMU measurements, while the outer loop runs at low gain with the visual input of the camera taking care of translation. The visual servo controller (outer loop) provides the attitude control (inner loop) the desired targets and outer loop ensures the stability of the system. An advantage of this approach is the possibility of reuse the IBVS scheme in other platforms since the low-level control of the specific material equipment of the aerial robot is not involved in it.

The paper **bourquardez\_2009** focuses in the design of kinematic controllers.

The use of zero and first-order image moments as visual features is considered inappropriate for aggressive maneuvers due to its lack of robustness for this case. For this reason, first-order spherical moments are introduced. Different control schemes are proposed in the paper. First, one controller using perspective projections is able of regulating the translation of the system thanks to the decoupled relationship between image and task space obtained. For the case of the camera image plane being parallel to the target plane, it is possible to control the translation of the robot independently of its rotation in a way equivalent to PBVS but without any pose estimation involved. The target depth is introduced as an initial data, so no depth estimation is performed. The strategy is easy to implement and provides a good result within the assumed geometrical limitation.

To overcome the limitations of the simple model presented, the work introduces order control schemes that include spherical projections as visual features and non-linear control techniques that improve the behavior for the case of aggressive maneuvers, although still around the hovering position.

In **ceren\_image\_2012** present a kinematic controller based on a Hybrid Visual Servoing scheme. The desired velocities are computed with the visual information ,

The authors in **asl\_vision-based\_2015** propose two improvements. For most of the controllers in the literature the camera velocity is required. This comes usually from IMU estimations, but due to the usual lack of GPS of the robots the noisy measurements of the IMU may not be enough. The paper proposed a visual flow and non-linear observer strategy to overcome this inconvenience. Secondly, in the standard formulation of the Jacobian matrix a estimation of the visual feature depth is necessary, something which is expensive

to include in the controller. This is usually solved thanks to the use of image moments and a predefined target depth. The authors include here a controller to deal with this uncertainty.

### 2.2.2 Visual Servoing for Aerial Manipulators

In this section a perspective of Visual Servoing applied to aerial manipulators is presented. The main available literature is analyzed and the common approaches followed in it highlighted. At the end of the section, Table 2.1 summarizes the strategies commented attending with regard to the main characteristics of each method.

Some publications related to the University of Pennsylvania GRASP Laboratory<sup>2</sup> (see **thomas\_toward\_2014** and **thomas\_visual\_2016**) have studied the vision-based localization and servoing of quadrotors in grasping and perching tasks. However, the emphasis of these publications lays on the generation of dynamically-feasible trajectories in the image space, thus second order system control is performed instead of the most common kinematic control strategy. These vehicles are not manipulators in the sense of the rest of the approaches presented in this section, since the main task here is hanging from structures and grasping targets by means of aggressive, thus dynamical, maneuvers. In addition to that, the actuator used is not a high-DOF serial manipulator, but a 1 DOF gripper.

In the last years, a concrete aerial manipulator architecture has been popularized, for example in the context of the European projects ARCAS<sup>3</sup> and AEROARMS<sup>4</sup>. These aerial manipulators have usually the task of collecting an structural element from its initial position and fly it to a final position, where the element is used to assemble a strut structure. The main configuration of such a robot is an under-actuated rotary-wing aircraft (usually a quadrotor) and a robotic serial manipulator arm. Different degrees of freedom (DOF) are used for the arm and different camera placements are considered.

The usual implementation considers the simultaneous control (at the velocity level) of the mobile platform (i.e. quadrotor) and the manipulator for such a grasping task. Since the sum of the 4 DOF of a quadrotor plus the multiple-DOF of a serial manipulator leads to a redundant system, the possibility of

---

<sup>2</sup><https://www.grasp.upenn.edu>

<sup>3</sup><http://www.arcas-project.eu>

<sup>4</sup><http://www.arcas-project.eu>

choosing degrees of freedom is used to realize different subtasks (e.g. joint limit reaching prevention). The Visual Servoing controller chosen generates velocity inputs both for the manipulator joints (i.e.  $\dot{\mathbf{q}}$ ) and for the quadrotor (i.e. translational velocity  $\mathbf{v}$  and rotational velocity  $\omega_z$ ). The use of a weighted pseudo-inverse allows to favor the control of the mobile platform when the distances to the target are bigger and increase the manipulator mobility when it is close to the target.

**mebarki\_image-based\_2014** propose a quadrotor equipped with a 5 DOF arm. Traditional Visual Servoing distinguishes between two different classes of camera configuration: eye-to-hand (fixed in the workspace) and eye-on-hand (mounted on the mobile platform). In this paper a new configuration for the camera is presented, called onboard-eye-to-hand, i.e. the camera is placed on-board of the robot while it observes the manipulator. In this way, the manipulator can accomplish large rotations while the target is not left out of the camera field of view, as happens in the eye-in-hand configuration. Furthermore, for the case of eye-in-hand configuration, during assembly tasks the manipulator end-effector can contact or impact with objects and damage or obstruct the camera. Thanks to the onboard-eye-to-hand camera configuration the paper is able to introduce a variation of the IBVS approach, called Self Visual Servoing (SVS). Where the error nullified comes directly from the image itself (hence the adjective self) and there is no need for a target image. The servo controller implemented has two different tasks. The main task is positioning the feature points at a target position on the target object and the second one the end-effector motion. The error formulation decouples both tasks and a weighted pseudo-inverse is used to provide a different gain for the arm joints rates  $\dot{\mathbf{q}}$  than for the UAV velocities  $\mathbf{v}$  and  $\omega_z$  control.

Image moments as features for the Visual Servoing are proposed in **mebarki\_exploiting\_2013** for the previous system. Furthermore, aerial manipulators have to cope with the change of the center of mass during flight due to the effect of suspended loads (see **palunko\_2012**). To achieve this behavior, low-level attitude controllers are usually designed to compensate this effects using Cartesian impedance control (see **lippiello\_impedance\_2012**) or an adaptive control approach is followed. In this case, the system includes a controller to reduce dynamic effects by vertically aligning the arm center of gravity to the multirotor gravitational vector, along with one that keeps the arm close to a desired configuration of high manipulability and avoiding arm joint limits. In **mebarki\_cross-coupled\_2014** the author completes the robot with a nonlinear low-level controller that thanks

to a integral approach allows the inclusion of the dynamic coupling of the UAV and the robotic arm, while the control of the system through the velocities provided by the IBVS high-level is maintained.

In this case the source (see **danko\_evaluation\_2014**) includes as host a gantry used to emulate an UAV and a 6 DOF manipulator with an end-effector mounted camera (i.e. eye-in-hand). Coordination of redundant degrees of freedom by means of partitioned control. Visual servoing is used to drive the end-effector pose relative to a target thanks to the use of feature points and their desired positions.

**lippiello\_hybrid\_2016** uses a hybrid-control framework to take advantage of the main benefits of both IBVS and PBVS schemes to control a octorotor with a 6 DOF arm. Kinematic redundancy of the end-effector is used to accomplish secondary tasks and lead by a hierarchical task-composition algorithm, in conjunction with a smooth activation mechanism for the tasks.

DLR's work within ARCAS project (see **laiacker\_high\_2016**) uses of an helicopter and a 7 DOF manipulator. The helicopter is a bigger robot when compared to the rest of the systems, with more than 1 m from manipulator to center of gravity. Influence of the arm movement is significant to the helicopter flight and is actively compensated by the robot controller by means of a coordinated control of both elements. The paper discusses performance and accuracy in aerial manipulation, where the time it takes between the measurement of a position difference and its compensation using the manipulator or the flying platform is the main factor. Additionally, the work presents a multi-marker approach to compensate the possible occlusion of the target marker by the manipulator derived of the onboard-eye-to-hand camera configuration.

A combination of kinematic and dynamic models to develop a passivity-based adaptive controller which can be applied on both position and velocity control is proposed in **kim\_vision-guided\_2016**. Position control is used for waypoint tracking and landing, while velocity control is triggered for target servoing. The robot is a quadrotor with a 3 DOF arm and an eye-in-hand camera. The work uses IBVS with image moments as visual features and tries to solve two problems of the method when applied to aerial manipulators (under-actuated system). Firstly, movements of the manipulator produce movement of the camera, thus making probable that the target object is taken out of its field of view. For this reason a fisheye camera is used, so it is possible to introduce a bigger field of view. Secondly, the under-actuation of the robot is corrected introducing a image modification method. Velocity weighting of the Jacobian



matrix in accordance of the situation is used for the simultaneous control of UAV and manipulator.

In **santamaria-navarro\_uncalibrated\_2017** redundant manipulation and hierarchical control law are combined with a new variation of the IBVS that does not need the camera parameters. The system establishes as primary task the avoidance of obstacles, as well as several secondary tasks. The visual servo strategy is used to drive the arm end-effector to a desired position and orientation by using a camera attached to it. The configuration used is a 4 DOF quadrotor, which is equipped with a 6 DOF robotic arm. In the common IBVS approaches, Jacobian or interaction matrix, which relates the camera velocity with the image feature velocities, depends on a priori knowledge of the intrinsic camera parameters. The paper presents a variation of IBVS called Uncalibrated IBVS, the approach uses the barycenter of the features as control points. The method recovers the coordinates of these control points and also the camera focal length, with this data a new formulation of the Jacobian is constructed. The system also compensates by means of a hierarchical algorithm the position of the manipulator.

Reference	Vehicle	Arm's DOF	Camera Configuration	VS Type	Visual feature	Comment
thomas_toward_2014 and thomas_visual_2016	quadrotor	1	eye-in-hand	IBVS	Cylinder parameters	agressive maneuvers
mebarki_image-based_2014	quadrotor	5	onboard-eye-to-hand	SVS	points (no target)	-
mebarki_exploiting_2013	quadrotor	5	onboard-eye-to-hand	SVS	perspective projection image moments (no target)	-
mebarki_cross-coupled_2014	quadrotor	5	onboard-eye-to-hand	SVS	points (no target)	low-lever controller for dynamic coupling robot-arm
danko_evaluation_2014	grantry	6	eye-in-hand	IBVS	points	gantry used to emulate an UAV
lippiello_hybrid_2016	octotor	6	onboard-eye-to-hand	Hybrid VS	points	hierarchical task-composition algorithm, smooth task activation
lailacker_high_2016	helicopter	7	onboard-eye-to-hand	IBVS	points	discusses performance and accuracy, multi-marker approach
kim_vision-guided_2016	quadrotor	3	eye-in-hand	IBVS	corrected perspective projection image moments	adaptive controller for both position and velocity, fisheye camera
santamaria-navarro_uncalibrated_2016	quadrotor	6	eye-in-hand	Uncalibrated IBVS	blobs' barycenters	hierarchical task-composition algorithm

**Table 2.1:** Overview of the different approaches for VS in flying manipulators

## 3 Software Requirements Specification and Structured Analysis

This chapter deals with the Software Requirement Specification (SRS) **IEEE8301998** and the Structured Analysis **SA\_Braune** of the system developed in this work. Thanks to these two procedures, the objectives that the system must fulfil and a decomposition of it into different functions are stated. This leads to a complete definition of the system.

The purpose of this work is to design a Visual Servoing controller to provide an underactuated aerial robot the commands necessary to reach a desired pose with respect to a target object.

The VS controller developed is to be integrated into the `hector_quadrotor` **2012simpar\_meyer** an underactuated aerial robot equipped with a monocular monochrome camera pointing downwards.

### 3.1 Software Requirement Specification

In this section the Software Requirement Specification **IEEE8301998** for the Visual Servoing controller developed in this thesis is presented. Using SRS helps to define the system that is being designed, tracking continuously that the product developed satisfies the needs of the user. Only when every requirement stated therein is fulfilled the implementation would be completed.

#### 3.1.1 Product Perspective

The VS controller is to be used with an aerial robotic system based on the ROS framework. From the perspective of the robotic system, the VS controller subsystem will appear as a ROS node which publishes control commands through a ROS topic to the rest of the system.

The used aerial robotic system is the `hector_quadrotor`<sup>1</sup> model **2012simpar\_meyer** (TODO: Add diagram).

---

<sup>1</sup>[wiki.ros.org/hector\\_quadrotor](http://wiki.ros.org/hector_quadrotor)

The subsystem developed here is to interact with the camera hardware of the robot, a monocular monochrome camera pointing downwards (TODO: Add hardware). The output of the subsystem are the control inputs of the aerial robot dynamics (TODO: Add which are the quadrotor control inputs), this inputs interact with the inner control loop for the attitude and outer control loop for the position already implemented in the robotic system (TODO: Position loop is not related to vs system, but can be useful for benchmark).

#### 3.1.2 User Characteristics

The product developed in this thesis will be used as part of a ROS-based system, thus the expected user is a designer willing to implement a VS control strategy for his robotic system. The user should be familiarized with the ROS framework and the system will need the structure and interfaces of any standard ROS product.

#### 3.1.3 Assumptions and Dependencies

The software has been tested on the following platforms, forward or backward support is not guaranteed on a different set-up.

- ROS version: ROS Indigo<sup>2</sup>
- Operating System: Ubuntu 14.04<sup>3</sup> Trusty Tahr, 64 bit

#### 3.1.4 Functional Requirements

The functional requirements describe what the system must do to complete the overall task:

- F1: *Give visual servoing control input.* Control input based on image data so that the aerial robot comes closer to the target pose. Control as a difference on the image features, no pose estimation.
- F2: *Tell user when the target pose is achieved.* The system must be able of telling the user whether the target pose has been already achieved or not.

---

<sup>2</sup><http://wiki.ros.org/indigo>

<sup>3</sup><http://releases.ubuntu.com/14.04/>

### 3.1.5 Other Requirements

- A1: All components are working reliably.
- A2: The software is sufficiently fast, modular and modifiable.
- A3: The implementation is transparent and comprehensible.
- A4: Control inputs must provide stable and smooth flight manoeuvres.
- A5: Robot must be able to start from different initial positions.
- A6: Algorithm must be fast enough to allow real time control of the aerial robot.
- A7: The implementation should follow the style guide of ROSROS\_\_Style

### 3.1.6 General Constraints

- The environment must be sufficiently illuminated for the camera to work.
- The target pose must be provided by a sufficient number of features (TODO: How many?) in form of a 2D code (TODO: At least in the first version).
- The target must be always in the field of view of the camera, so features can be extracted and control input computed.
- Testing computer is a MacBook Pro<sup>4</sup> (Early 2015) with 2.7 GHz Intel Core i5 processor, 8 GB 1867 MHz DDR3 memory and Intel Iris Graphics 6100 1536 MB graphics. Linux OS is run using Oracle VM VirtualBox<sup>5</sup> (Version 5.1.14 r112924) with 5 GB base memory and two processors.

## 3.2 Structured Analysis

---

<sup>4</sup>[https://everymac.com/systems/apple/macbook\\_pro/specs/macbook-pro-core-i5-2.7-13-early-2015-retina-display-specs.html](https://everymac.com/systems/apple/macbook_pro/specs/macbook-pro-core-i5-2.7-13-early-2015-retina-display-specs.html)

<sup>5</sup>[www.virtualbox.org](http://www.virtualbox.org)

## **4 Visual Servoing Algorithm Description**

## **5 Implementation of the Visual Servoing Controller**

## **6 Final Results and Conclusions**



## **7 Future Work**

## Selbstständigkeitserklärung

Hiermit versichere ich, Pablo Rodríguez Robles, geboren am 28.02.1996 in León, Spain, dass ich die vorliegende Bachelor Thesis zum Thema

*Image Based Visual Servoing for Aerial Robot*

ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Bei der Auswahl und Auswertung des Materials sowie bei der Herstellung des Manuskripts habe ich Unterstützungsleistungen von folgenden Personen erhalten:

*Dipl.-Ing. Chao Yao*

Weitere Personen waren an der geistigen Herstellung der vorliegenden Bachelor Thesis nicht beteiligt. Mir ist bekannt, dass die Nichteinhaltung dieser Erklärung zum nachträglichen Entzug des Diplomabschlusses (Masterabschlusses) führen kann.

Dresden, den 27.03.2018

.....  
Unterschrift