Flying Calligrapher: Contact-Aware Motion and Force Planning and Control for Aerial Manipulation

Xiaofeng Guo*[1], Guanqi He*[1], Jiahe Xu[1], Mohammadreza Mousaei[1], Junyi Geng[2],
Sebastian Scherer[1], Guanya Shi[1]

***Abstract*—Aerial manipulation has gained interest in completing high-altitude tasks that are challenging for human workers, such as contact inspection and defect detection, etc. Previous research has focused on maintaining static contact points or forces. This letter addresses a more general and dynamic task: simultaneously tracking time-varying contact force in the surface normal direction and motion trajectories on tangential surfaces. We propose a pipeline that includes a contact-aware trajectory planner to generate dynamically feasible trajectories, and a hybrid motion-force controller to track such trajectories. We demonstrate the approach in an aerial calligraphy task using a novel sponge pen design as the end-effector, whose stroke width is positively related to the contact force. Additionally, we develop a touchscreen interface for flexible user input. Experiments show our method can effectively draw diverse letters, achieving an IoU of 0.59 and an end-effector position (force) tracking RMSE of 2.9 cm (0.7 N). Website: https://xiaofeng-guo.github.io/flying-calligrapher/**

***Index Terms*—Aerial Systems: Applications; Aerial Systems: Mechanics and Control; Integrated Planning and Control**

I. Introduction

Uncrewed Aerial Vehicles (UAVs) have rapidly developed in recent years, with applications ranging from photography [1], [2] to package delivery [3], [4]. High-altitude tasks, which are often challenging and time-consuming for human workers and sometimes require active interaction with the environment, have expanded the requirements for UAV operations beyond just passive roles. Aerial interaction, or aerial manipulation, where UAVs intentionally interact with the external environment, has gained increased attention. Such a shift has sparked greater interest in UAVs for tasks such as contact defect inspection, cleaning, or maintenance at high altitudes. The key challenge lies in the nonlinear floating-based disturbance [5], [6] as well as the coupling between the flying vehicle and the attached manipulator.

There are many research efforts exploring the aerial manipulation in different scenarios, such as pushing a target [7], turning a valve [8], opening a door [9], drilling [10], screwing [11], cleaning [12], contact-based inspection [13], [14], etc. Uncrewed Aerial Manipulators (UAMs) in these scenarios often need to track a reference contact force or motion. However, most existing work focuses on specific tasks, such as maintaining a constant force or tracking discrete force targets at different locations. A general solution for tracking continuous, time-varying motion and force trajectories is lacking. On the other hand, there are many high-altitude applications that need this capability. For instance, in cleaning, the UAM's end-effector must slide on surfaces while adjusting to varying motion and contact force based on dirtiness. In carving or writing, the UAM must follow continuous, time-varying force and motion trajectories to accurately reflect stroke thickness. Relying solely on discrete force target tracking is too time-consuming and cumbersome.

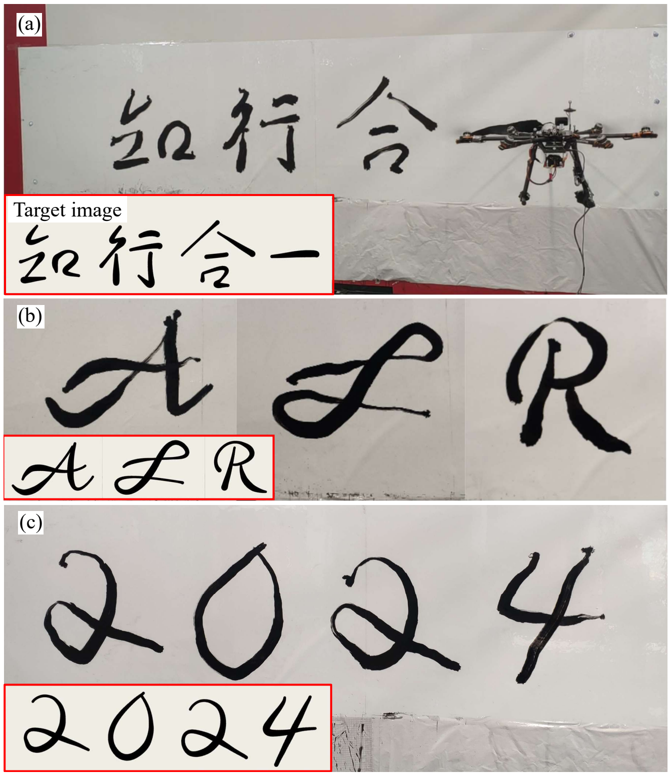

Fig. 1. (a) Our aerial manipulator is writing a Chinese calligraphy artwork on the wall. (b) "AIR" and (c) "2024" written by our system.

We aim to achieve general time-varying contact force and motion tracking to enable a wide range of aerial manipulation tasks. In particular, we target aerial calligraphy as a representative application to demonstrate the overall performance due to its high precision requirements on both motion and force tracking, and its superior visualization. Different from the

This work was supported by CMU Manufacturing Futures Institute (MFI).

*Equal Contribution.

[1]Xiaofeng Guo, Guanqi He, Jiahe Xu, Mohammadreza Mousaei, Sebastian Scherer, and Guanya Shi are with the School of Computer Science, Robotics Institute, Carnegie Mellon University, Pittsburgh PA 15213, USA. Email: {xguo2, guanqihe, jiahex, mmousaei, basti, guanyas}@andrew.cmu.edu

[2]Junyi Geng is with the College of Engineering, Department of Aerospace Engineering, Pennsylvania State University, University Park, PA, 16802, USA. Email: jgeng@psu.edu

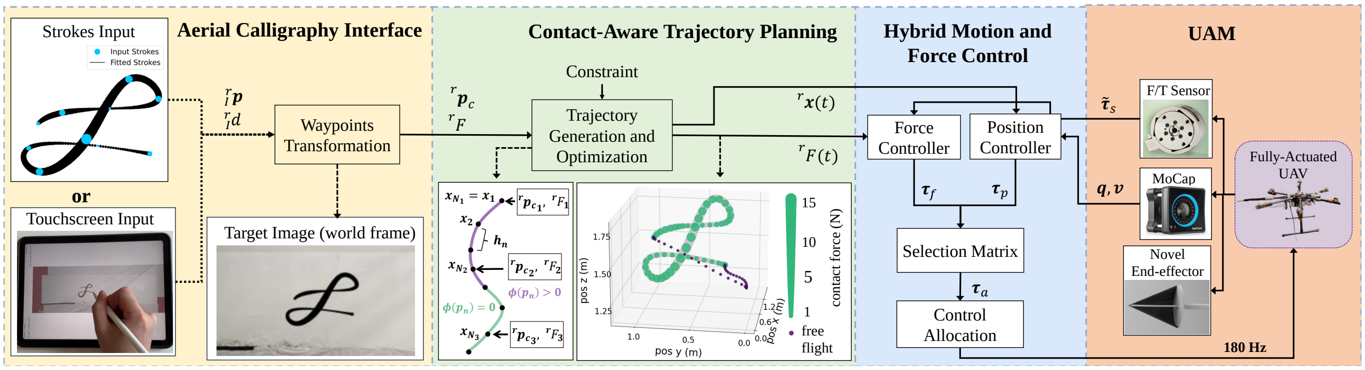

Fig. 2. Pipeline of the proposed system. Users either manually define the target strokes or draw the target letter through a touchscreen interface. Then, the trajectory planning module generates a dynamic-feasible motion-force trajectory. A hybrid motion and force controller is designed for the UAM to track the reference trajectory. Finally, a novel end-effector pen is designed for the UAM to draw the target letters.

simple aerial writing task, this task requires the UAM to draw letters with dynamically varying linewidth. It is challenging due to several reasons. The trajectory, including both the motion and contact force, must be smooth and dynamically feasible to ensure both feasibility and artistic quality. Unlike discrete force holding at different locations, the friction caused by the sliding with continuous time-varying contact forces during drawing must be properly handled by the controller. Additionally, the end-effector must be designed to reflect time-varying motion and contact force for proper visualization.

This paper develops a contact-aware hybrid motion-force planning and control strategy for aerial calligraphy using a fully-actuated hexarotor aerial manipulator. The system consists of three main components: a trajectory planning module, a hybrid control module, and a real-time user input interface via touchscreen. The touchscreen interface allows users to manually input or draw the target letter using an iPad. The trajectory planning module then generates a dynamically feasible motion and contact force trajectory, while the control module ensures the UAM accurately tracks these references to draw the letters. Additionally, a novel sponge pen end-effector is designed to visualize the drawn letters with varying linewidths based on contact force. In summary, the main contributions of this work are:

- We propose a contact-aware trajectory planning algorithm to generate dynamically feasible contact force and motion trajectory in the contact plane.
- We develop a contact-aware hybrid motion-force control algorithm to enable the UAM to track the continuous time-varying contact force and motion reference simultaneously, while compensating for friction force.
- We develop a system and pipeline for a novel aerial calligraphy task.

II. RELATED WORKS

In aerial manipulation tasks, UAMs often need to track reference contact force or motion trajectories during contact with external objects. To generate the reference trajectory, the existing works mainly focused on motion planning and treated the interaction force between the UAM and the environment as disturbances [15], [16], [17]. As for contact force estimation, researchers have developed contact mechanics models, such as quasi-static or spring-damper models [18], [19]. Some studies also mounted F/T sensors on UAMs to directly measure the contact force for feedback [20], [21].

Aerial writing, which requires the UAM to control both motion and contact force during the task, is one common task the researcher focuses on. Tzoumanikas et al. [22] developed a hybrid motion-force MPC as the trajectory generator and controller to draw letters with constant linewidth. They obtained the contact force using a linear spring model for vehicle dynamics. Lanegger et al. [23] designed a compliant Gough-Stewart mechanism for line drawing, improving stability and precision with multiple contact points. They equipped three omni-wheels with the end-effector and modeled the whole structure as a three-wheeled omnidirectional ground robot for precise control.

However, among all these works, most of them either track only a reference motion trajectory [20], [22], a set of discrete constant force targets at different locations [21], or time-varying force but without motion [24]. Our goal is to achieve general time-varying contact force and motion tracking to enable various aerial manipulation tasks, such as aerial calligraphy with strokes of varying linewidths. Unlike many studies that consider only normal contact forces, we utilize the full contact model in this work to predict the contact wrench (both normal force and friction force). This model is used in both trajectory planning to generate dynamically feasible trajectories and in hybrid motion-force control with contact wrench compensation. Additionally, our method treats execution time also as a decision variable to be optimized in trajectory optimization [25], improving UAM task efficiency compared to algorithms that predefine the execution time [15], [26].

III. PROBLEM DEFINITION AND METHOD OVERVIEW

A. Problem Definition

This work aims to simultaneously track motion and contact force for UAM during aerial interaction. The problem is defined as follows: control the UAM's end-effector to navigate through a sequence of contact points on a surface with a

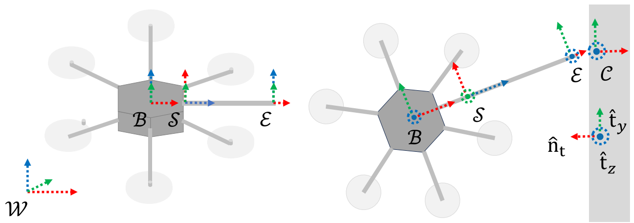

Fig. 3. The coordinate frames used in this paper. Specifically, $\mathcal{W}$, $\mathcal{B}$, $\mathcal{E}$, $\mathcal{S}$, and $\mathcal{C}$ stand for the world, vehicle body, end-effector, arm, and contact frame, respectively.

known shape, while applying the desired normal contact force at each contact point. Specifically, we focus on the aerial calligraphy task. Given a list of waypoints in the image space, each specifying the position and linewidth of a point, the goal is to control the UAM to accurately draw the letter.

B. System and Pipeline Overview

Our UAM platform is adopted from our previous works [13], which is a fully-actuated hexarotor equipped with a one-link manipulator. An F/T sensor is mounted at the base of the manipulator. The flying calligrapher system pipeline is shown in Fig. 2 and consists of four components. First, we develop an aerial calligraphy interface allowing users to specify the target image by either manually defining waypoints or drawing the letter through a touchscreen interface. This module then converts the user input to sparse waypoints in the task space, including the contact point position and magnitude of the normal contact force. Second, based on the sparse waypoints, we develop a contact-aware trajectory planner that generates a dynamically feasible motion-force trajectory for the UAM to track. Subsequently, a hybrid motion-force controller is designed so that the UAM can follow the reference trajectory. Finally, we design a novel end-effector for the UAV to draw the target letters.

C. Coordinate Frames and Notations

We define several coordinate frames, as shown in Fig. 3. The world inertial frame $\mathcal{W}$ is an east-north-up frame with the origin at the UAV's takeoff position. The body frame $\mathcal{B}$ originates at the vehicle's center of mass, with axes pointing front, left, and up relative to the vehicle. The end-effector frame $\mathcal{E}$ originates at the end-effector, with axes pointing front, left, and up relative to the vehicle. The arm frame $\mathcal{S}$ originates at the root of the manipulator where the F/T sensor is mounted, with axes aligned to the sensor. The contact frame $\mathcal{C}$ originates at the contact point, with axes aligned to the surface normal (towards inside), left, and up directions of the surface.

Vectors are denoted as bold lowercase symbols. Left-hand subscripts, e.g., ${}_{\mathcal{W}}\boldsymbol{v}$, indicate the coordinate representation in frame $\mathcal{W}$. $\boldsymbol{R}_a^b \in \mathbb{R}^{3\times3}$, $\boldsymbol{T}_a^b \in \mathbb{R}^{3\times1}$, $\boldsymbol{Ad}_{\boldsymbol{T}_a^b} \in \mathbb{R}^{6\times6}$ define the rotation, translation, and wrench (force and torque) transformation for vector from frame b to frame a, respectively [27]. We have: ${}_a\boldsymbol{v} = \boldsymbol{R}_a^b\,{}_b\boldsymbol{v}$. $\boldsymbol{0}_{a\times b}$ defines a $\boldsymbol{0}$ vector in $a \times b$ dimensions. $\boldsymbol{v}_{[i]}$ indicates the i^{th} element of $\boldsymbol{v}$; ${}^{r}(\cdot)$ denotes the reference quantity.

D. Aerial Manipulator Dynamics

We define the UAM end-effector position vector in the world frame as ${}_{\mathcal{W}}\boldsymbol{p} = [x, y, z]^\top$, the attitude with rotation matrix representation $\boldsymbol{R}_{\mathcal{W}}^{\mathcal{E}}$ and the angular velocity in the end-effector frame as $\boldsymbol{\omega}$. The pose and twist of the UAM end-effector are defined as $\boldsymbol{q} = [{}_{\mathcal{W}}\boldsymbol{p}, \boldsymbol{R}_{\mathcal{W}}^{\mathcal{E}}]$ and $\boldsymbol{v} = [{}_{\mathcal{W}}\dot{\boldsymbol{p}}^\top, \boldsymbol{\omega}^\top]^\top$.

The dynamic model of the hexarotor aerial manipulator at the end-effector can be derived following our previous work [28].

$$\boldsymbol{M}\dot{\boldsymbol{v}} + \boldsymbol{C}\boldsymbol{v} = \boldsymbol{Ad}_{\boldsymbol{T}_{\mathcal{E}}^{\mathcal{B}}}\,{}_{\mathcal{B}}\boldsymbol{\tau}_a + \boldsymbol{Ad}_{\boldsymbol{T}_{\mathcal{E}}^{\mathcal{C}}}\,{}_{\mathcal{C}}\boldsymbol{\tau}_c + \boldsymbol{Ad}_{\boldsymbol{T}_{\mathcal{E}}^{\mathcal{B}}}\boldsymbol{R}_{\mathcal{B}}^{\mathcal{W}}\,{}_{\mathcal{W}}\boldsymbol{g} \quad (1)$$

with inertia matrix $\boldsymbol{M} \in \mathbb{R}^{6\times6}$, centrifugal and Coriolis term $\boldsymbol{C} \in \mathbb{R}^{6\times6}$, gravity wrench $\boldsymbol{g} \in \mathbb{R}^6$, contact wrench $\boldsymbol{\tau}_c \in \mathbb{R}^6$, and control wrench $\boldsymbol{\tau}_a = [\boldsymbol{f}_a^\top, \boldsymbol{m}_a^\top]^\top \in \mathbb{R}^6$ from the UAM actuators, where $\boldsymbol{f}_a \in \mathbb{R}^3$ denotes the control force and $\boldsymbol{m}_a \in \mathbb{R}^3$ denotes the torque. Specifically, we have

$$\boldsymbol{M} = \begin{bmatrix} m\boldsymbol{R}_{\mathcal{E}}^{\mathcal{W}} & -m[{}_{\mathcal{E}}\boldsymbol{t}_{\mathcal{B}}^{\mathcal{E}}]_\times \\ m[{}_{\mathcal{E}}\boldsymbol{t}_{\mathcal{B}}^{\mathcal{E}}]_\times\boldsymbol{R}_{\mathcal{E}}^{\mathcal{W}} & \boldsymbol{R}_{\mathcal{E}}^{\mathcal{B}}\boldsymbol{J}\boldsymbol{R}_{\mathcal{B}}^{\mathcal{E}} - m[{}_{\mathcal{E}}\boldsymbol{t}_{\mathcal{B}}^{\mathcal{E}}]_\times[{}_{\mathcal{E}}\boldsymbol{t}_{\mathcal{B}}^{\mathcal{E}}]_\times \end{bmatrix} \quad (2)$$

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{0}_{3\times3} & -m[\boldsymbol{\omega}]_\times[{}_{\mathcal{E}}\boldsymbol{t}_{\mathcal{B}}^{\mathcal{E}}]_\times \\ \boldsymbol{0}_{3\times3} & [\boldsymbol{\omega}]_\times\boldsymbol{R}_{\mathcal{E}}^{\mathcal{B}}\boldsymbol{J}\boldsymbol{R}_{\mathcal{B}}^{\mathcal{E}} - m[{}_{\mathcal{E}}\boldsymbol{t}_{\mathcal{B}}^{\mathcal{E}}]_\times[\boldsymbol{\omega}]_\times[{}_{\mathcal{E}}\boldsymbol{t}_{\mathcal{B}}^{\mathcal{E}}]_\times \end{bmatrix} \quad (3)$$

Here m is the vehicle mass; $\boldsymbol{J}$ is the moment of inertia at the vehicle center of mass in the body frame; ${}_{\mathcal{E}}\boldsymbol{t}_{\mathcal{B}}^{\mathcal{E}}$ is the relative translation pointing from the tip of the end-effector to the vehicle body under the end-effector frame, and $[*]_\times$ is the skew-symmetric matrix associated with vector $*$.

The control wrench comes from the standard UAV rotor thrust and torque. As for the contact wrench, since the one-link manipulator is rigidly attached to the UAV contacting the surface through a point, only the reaction force is considered at the contact point without torque, i.e.

$${}_{\mathcal{C}}\boldsymbol{\tau}_c = [\boldsymbol{f}_c^\top, \boldsymbol{0}_{3\times1}^\top]^\top, \quad \boldsymbol{f}_c = \boldsymbol{f}_\perp + \boldsymbol{f}_{/\!/} \quad (4)$$

where $\boldsymbol{f}_\perp$ and $\boldsymbol{f}_{/\!/}$ represent the normal force and shear force respectively. Given the contact surface ϕ and the assumption that the end-effector makes a point-contact with the surface, we have the contact constraint $\phi(\boldsymbol{p}) = 0$. The contact axes can be represented by the surface normal vector $\hat{\boldsymbol{n}}_t$, and the in-plane basis vectors $\hat{\boldsymbol{t}}_y - \hat{\boldsymbol{t}}_z$ with

$$\hat{\boldsymbol{n}}_t = \frac{\nabla\phi(\boldsymbol{p})}{\|\nabla\phi(\boldsymbol{p})\|}, \hat{\boldsymbol{t}}_y = \frac{\boldsymbol{g}_{[1:3]} \times \hat{\boldsymbol{n}}_t}{\|\boldsymbol{g}_{[1:3]} \times \hat{\boldsymbol{n}}_t\|}, \hat{\boldsymbol{t}}_z = \frac{\hat{\boldsymbol{t}}_y \times \hat{\boldsymbol{n}}_t}{\|\hat{\boldsymbol{t}}_y \times \hat{\boldsymbol{n}}_t\|} \quad (5)$$

Therefore, the contact force can be expressed as

$$\begin{aligned} F &= \|\boldsymbol{f}_\perp\|, \quad \boldsymbol{f}_\perp = F\hat{\boldsymbol{n}}_t \qquad (6) \\ \boldsymbol{f}_{/\!/} &= -\mu F\left(\text{sign}(\dot{y})\hat{\boldsymbol{t}}_y + \text{sign}(\dot{z})\hat{\boldsymbol{t}}_z\right) \\ \boldsymbol{\alpha}(\boldsymbol{v}) &= \left[[\hat{\boldsymbol{n}}_t - \mu\text{sign}(\dot{y})\hat{\boldsymbol{t}}_y - \mu\text{sign}(\dot{z})\hat{\boldsymbol{t}}_z]^\top, \boldsymbol{0}_{3\times1}^\top\right]^\top \\ {}_{\mathcal{C}}\boldsymbol{\tau}_c &= F\boldsymbol{\alpha}(\boldsymbol{v}) \end{aligned}$$

where F is the magnitude of the normal force. Here, we apply the Coulomb model to estimate the friction force with the friction coefficient μ. The negative sign at the beginning represents that the friction is in the opposite direction of the motion and sign$(\cdot)$ denotes the sign function.

IV. CONTACT-AWARE TRAJECTORY PLANNING

This section introduces the proposed contact-aware trajectory planning algorithm. It takes M sparse waypoints $({}^{r}\boldsymbol{p}_{ci}, {}^{r}F_i)$, for $i \in 1, \cdots, M$ and contact surface ϕ as input, and outputs the state and action trajectories $\boldsymbol{x}(t)$, $\boldsymbol{\tau}_a(t)$ as the reference for the hybrid motion-force controller, where $\boldsymbol{x}(t) = [\boldsymbol{q}^\top(t), \boldsymbol{v}^\top(t), F(t)]^\top$, $t \in (0, T)$. The waypoints can include both contact points where ${}^{r}F_i > 0$ and non-contact points where ${}^{r}F_i = 0$. Notice that our algorithm also optimizes the total time, therefore timestamps are not included in the input waypoints. Different from other works that consider contact reaction wrench as a disturbance, we treat $\boldsymbol{\tau}_c(t)$ as a predictable state in the trajectory planning, and we also output it to the controller.

The trajectory planning algorithm is to find an N-step state and control sequences $\boldsymbol{x}_{1:N}$, $\boldsymbol{\tau}_{a1:N}$, which can be formulated as an optimization problem:

$$\min_{\substack{\boldsymbol{x}_{1:N}, \\ \boldsymbol{\tau}_{a1:N}, \\ h_{1:N}}} \sum_{n=1}^{N} \left(\|\boldsymbol{v}_n\|^2{}_{\boldsymbol{W}_v} + \|\boldsymbol{\tau}_{an}\|^2_{\boldsymbol{W}_\tau} + \|\delta\boldsymbol{\tau}_{an}\|^2_{\boldsymbol{W}_{d\tau}} + \gamma \right) h_n \tag{7a}$$

$$s.t. \quad \boldsymbol{x}_{n+1} = f_{dyn}(\boldsymbol{x}_n, \boldsymbol{\tau}_{an}, h_n) \tag{7b}$$

$$0 < h_n \leq \bar{h} \tag{7c}$$

$$\boldsymbol{p}_{N_i} = {}^{r}\boldsymbol{p}_{ci}, \quad F_{N_i} = {}^{r}F_i, \ \forall i \in 1, \cdots, M \tag{7d}$$

$$\phi(\boldsymbol{p}_n) \begin{cases} = 0, \text{if } {}^{r}F_{N_i}{}^{r}F_{N_{i+1}} > 0 \\ > 0, \text{if } {}^{r}F_{N_i}{}^{r}F_{N_{i+1}} = 0 \end{cases} \quad \forall n \in [N_i, N_{i+1}] \tag{7e}$$

$$g(\boldsymbol{x}_n) = 0 \tag{7f}$$

$$\begin{aligned} &\underline{\boldsymbol{\tau}}_a \leq \boldsymbol{\tau}_{an} \leq \bar{\boldsymbol{\tau}}_a, \quad \underline{\dot{\boldsymbol{\tau}}}_a \leq \delta\boldsymbol{\tau}_{an} \leq \bar{\dot{\boldsymbol{\tau}}}_a \\ &\underline{\boldsymbol{v}} \leq \boldsymbol{v}_n \leq \bar{\boldsymbol{v}}, \quad \underline{\dot{F}} \leq \delta F \leq \bar{\dot{F}} \end{aligned} \tag{7g}$$

where $\|\boldsymbol{x}\|_{\boldsymbol{W}} := \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{x}$ denotes the $\boldsymbol{W}$-norm of $\boldsymbol{x}$. The cost function in (7a) penalizes the twist $\boldsymbol{v}_n$, control wrench $\boldsymbol{\tau}_{an}$, changing rate of the wrench $\delta\boldsymbol{\tau}_{an}$ and time-interval between two steps h_n. $\boldsymbol{W}_v$, $\boldsymbol{W}_\tau$, $\boldsymbol{W}_{d\tau}$ are three diagonal weighting matrices and γ is the scalar of time penalty.

(7b) represents the discrete-time dynamics of the system (1), derived using RK4 discretization. Since timestep h_n is also an optimization variable, we impose an upper limit $\bar{h}$ on the time interval for each step (7c) to ensure dynamic accuracy. The value of $\bar{h}$ is determined based on the trajectory length and the total number of steps N, and is fine-tuned during the experiments.

(7d) ensures the UAM tracks waypoints by imposing the vehicle sequentially pass through the i^{th} waypoint at step N_i. Due to the upper limit for the time interval h_n, the optimization may become infeasible if there are insufficient steps between waypoints. Conversely, too many steps will cause a significant computational burden. We pre-allocate the step-index N_i in proportion to the distance between adjacent waypoints to reduce the optimization time while ensuring feasibility. We set $N_1 = 1$ and $N = N_M$.

(7e) ensures the end-effector contacts the target surface during the segment between two contact waypoints and ensures the end-effector does not make contact with the surface during other segments.

(7f) provides the task-related constraint. Here, in the aerial calligraphy, to maintain the end-effector perpendicular to the wall for better drawing performance, the desired UAM orientation is constructed by (5)

$$g(\boldsymbol{x}) := \boldsymbol{R}^{\mathcal{B}}_{\mathcal{W}} \ominus {}^{r}\boldsymbol{R}^{\mathcal{B}}_{\mathcal{W}}, \quad {}^{r}\boldsymbol{R}^{\mathcal{B}}_{\mathcal{W}}(\boldsymbol{p}) = [\hat{\boldsymbol{t}}_z, \hat{\boldsymbol{t}}_y, -\hat{\boldsymbol{n}}_t] \tag{8}$$

where $\ominus$ denotes the subtraction operation in the $SO(3)$ manifold. Additional safety and feasibility constraints for vehicle control wrench, velocity and contact force are added as (7g), where $\underline{\boldsymbol{a}}$ and $\bar{\boldsymbol{a}}$ represents the lower limit and upper limit of the variable $\boldsymbol{a}$.

Since the time intervals obtained from problem (7a) may be non-uniform, we apply linear interpolation to the planning results to ensure the control and state sequences are consistent with the controller frequency. Given $\boldsymbol{x}_{1:T}, \boldsymbol{\tau}_{a1:T}, h_{1:T}$, for $t \in [T_j, T_{j+1}]$, and $T_j = \sum_{n=1}^{j} h_n$,

$$\boldsymbol{x}(t) = \lambda\boldsymbol{x}_j + (1-\lambda)\boldsymbol{x}_{j+1}, \ \boldsymbol{\tau}_a(t) = \lambda\boldsymbol{u}_j + (1-\lambda)\boldsymbol{\tau}_{aj+1}$$

with $\lambda = (t - T_j)/(T_{j+1} - T_j)$. We also calculate the model predicted contact wrench $\boldsymbol{\tau}_c(t)$ from (6) for later use. In summary, the planning algorithm outputs ${}^{r}\boldsymbol{x}(t)$, ${}^{r}\boldsymbol{\tau}_a(t)$, and ${}^{r}\boldsymbol{\tau}_c(t)$ for the hybrid motion-force controller.

V. HYBRID MOTION-FORCE CONTROL

This section discusses the design of the hybrid motion-force controller. Based on our previous work [13], we enhance the controller by incorporating contact wrench models, including the dynamics of contact force, friction force, and friction-induced torque.

A. Contact Wrench Estimation

Unlike the free-flight UAV, the primary challenge of UAM is to compensate the contact wrench $\boldsymbol{\tau}_c$ during aerial interaction. We estimate it by fusing the sensor readings into the analytic model.

As described in (4), ${}_{\mathcal{C}}\boldsymbol{\tau}_c$ consists of both the normal and shear forces. Since we have an F/T sensor at the origin of frame $\mathcal{S}$, it allows us to estimate ${}_{\mathcal{C}}\boldsymbol{\tau}_c$. Specifically, for the aerial calligraphy task, given the planned trajectory ${}^{r}\boldsymbol{x}$, we obtain ${}^{r}_{\mathcal{C}}\boldsymbol{\tau}_c = {}^{r}F^{r}\boldsymbol{\alpha}({}^{r}\boldsymbol{v})$. On the other hand, F/T sensor provides the measurement ${}_{\mathcal{S}}\tilde{\boldsymbol{\tau}}_s$ for the wrench at the origin of frame $\mathcal{S}$ (denote as ${}_{\mathcal{S}}\boldsymbol{\tau}_s$). In addition, it is obvious to see that the relation of the ${}_{\mathcal{S}}\boldsymbol{\tau}_s$ and the contact wrench can be expressed as ${}_{\mathcal{B}}\boldsymbol{\tau}_c = \boldsymbol{Ad}_{\boldsymbol{T}^{\mathcal{S}}_{\mathcal{B}}}{}_{\mathcal{S}}\boldsymbol{\tau}_s$. Therefore, the contact normal force can be estimated by fusing the sensor measurement with the contact force model

$$\hat{F} = \left[\boldsymbol{Ad}_{\boldsymbol{T}^{\mathcal{C}}_{\mathcal{B}}}{}^{r}\boldsymbol{\alpha}\right]^{+} \boldsymbol{Ad}_{\boldsymbol{T}^{\mathcal{S}}_{\mathcal{B}}}{}_{\mathcal{S}}\tilde{\boldsymbol{\tau}}_s \tag{9}$$

where $[\cdot]^{+}$ denotes the pseudo-inverse of a matrix. Furthermore, under hover or zero tilt condition of the fully-actuated

UAM and given the specific flat vertical wall, $\boldsymbol{R}^{\mathcal{C}}_{\mathcal{B}} \approx \boldsymbol{I}$. Thus, the estimate can be simplified as $\hat{F} = [\boldsymbol{R}^{\mathcal{S}}_{\mathcal{B}}\,{}_{\mathcal{S}}\boldsymbol{\tau}_{s[1:3]}]_{[1]}$. The corresponding contact wrench estimates can be obtained as

$$ {}_{\mathcal{C}}\hat{\boldsymbol{\tau}}_c = \hat{F}\boldsymbol{\alpha}({}^r\boldsymbol{v}) \tag{10} $$

Here, we use reference velocity ${}^r\boldsymbol{v}$ instead of measured velocity for robustness considerations.

B. Hybrid Motion-Force Controller

The fully-actuated UAM is able to control the translation and orientation independently. We develop a contact-aware hybrid motion-force controller based on our previous work [13] for the UAM to track the planned trajectory ${}^r\boldsymbol{x}(t)$.

The control wrench $\boldsymbol{\tau}_a$ combines the output of the motion tracking controller $\boldsymbol{\tau}_p$ and wrench tracking controller $\boldsymbol{\tau}_f$ as follows:

$$ {}_{\mathcal{B}}\boldsymbol{\tau}_a = \boldsymbol{R}^{\mathcal{C}}_{\mathcal{B}}((\mathbf{I}_{6\times 6} - \boldsymbol{\Lambda})\boldsymbol{R}^{\mathcal{B}}_{\mathcal{C}}\,{}_{\mathcal{B}}\boldsymbol{\tau}_p + \boldsymbol{\Lambda}\,{}_{\mathcal{C}}\boldsymbol{\tau}_f) \tag{11} $$

where $\boldsymbol{\Lambda} = \text{blockdiag}[\delta, 0, 0, 0, 0, 0] \in \mathbb{R}^{6\times 6}, \delta \in \{0, 1\}$. Intuitively, the matrix $\boldsymbol{\Lambda}$ selects direct force control commands and leaves the complementary subspace for the motion control. δ is the force motion control switch based on ${}^rF$. δ is set to 0 when ${}^rF = 0$, corresponding to the pure motion control, and δ is set to 1 when ${}^rF > 0$, indicating the force control along the normal direction of the surface.

For motion control, a PID controller is designed to directly control the end-effector motion.

$$ \begin{aligned} {}_{\mathcal{B}}\boldsymbol{\tau}_p =& {}_{\mathcal{B}}{}^r\boldsymbol{\tau}_a + \boldsymbol{Ad}_{\boldsymbol{T}^{\mathcal{C}}_{\mathcal{B}}}({}_{\mathcal{C}}{}^r\boldsymbol{\tau}_c - {}_{\mathcal{C}}\hat{\boldsymbol{\tau}}_c) \\ & - \boldsymbol{K}_q\boldsymbol{e}_q - \boldsymbol{K}_v\boldsymbol{e}_v - \boldsymbol{K}_i\int \boldsymbol{e}_q dt \end{aligned} \tag{12} $$

where

$$ \begin{aligned} &\boldsymbol{e}_q = [\boldsymbol{e}_p^\top, \boldsymbol{e}_R^\top]^\top,\ \boldsymbol{e}_v = [\dot{\boldsymbol{e}}_p^\top, \boldsymbol{e}_\omega^\top]^\top,\ \boldsymbol{e}_p = \boldsymbol{R}^{\mathcal{W}}_{\mathcal{B}}({}_{\mathcal{W}}\boldsymbol{p} - {}_{\mathcal{W}}{}^r\boldsymbol{p}) \\ &\boldsymbol{e}_R = \frac{1}{2}({}^r{\boldsymbol{R}^{\mathcal{B}}_{\mathcal{W}}}^\top \boldsymbol{R}^{\mathcal{B}}_{\mathcal{W}} - \boldsymbol{R}^{\mathcal{B}}_{\mathcal{W}}\,{}^r{\boldsymbol{R}^{\mathcal{B}}_{\mathcal{W}}}^\top)^\vee,\ \boldsymbol{e}_\omega = \boldsymbol{\omega} - {}^r\boldsymbol{\omega} \end{aligned} \tag{13} $$

${}^r_{\mathcal{B}}\boldsymbol{\tau}_a$ and ${}^r_{\mathcal{C}}\boldsymbol{\tau}_c$ come from the planning. Here $(\cdot)^\vee$ is to extract a vector from a skew-symmetric matrix and $\boldsymbol{K}_q, \boldsymbol{K}_v, \boldsymbol{K}_i$ are the positive definite tunable controller gain matrices. Here ${}_{\mathcal{C}}\hat{\boldsymbol{\tau}}_c$ is the key novelty of the proposed controller that we incorporate contact wrench models.

As for wrench tracking, we design an impedance controller to ensure that the end-effector maintains the desired wrench. Given the contact force reference F^r and the estimated contact normal force (9), the controller is designed as:

$$ \begin{aligned} e_f &= \hat{F} - {}^rF, \quad e_x = \boldsymbol{e}_{q[1]} \\ F_f &= {}^rF - m_0 m_d^{-1}(K_e e_x + D_e \dot{e}_x) \\ &\quad - K_{f,p}e_f - K_{f,i}\int e_f dt \\ &= {}^rF - K_{e,p}e_x - K_{e,d}\dot{e}_x - K_{f,p}e_f - K_{f,i}\int e_f dt \end{aligned} \tag{14} $$

with m_0, m_d the actual and desired vehicle mass, respectively, $K_{e,p}$ is the normalized stiffness, $K_{e,d}$ is the normalized damping. Then, the control wrench ${}_{\mathcal{C}}\boldsymbol{\tau}_f = [F_f, \mathbf{0}_{1\times 5}]^\top$ and ${}_{\mathcal{W}}\boldsymbol{\tau}_p$ will be passed through (11) to get the total control wrench. Finally, ${}_{\mathcal{B}}\boldsymbol{\tau}_a$ is distributed to each rotor through the control allocation process.

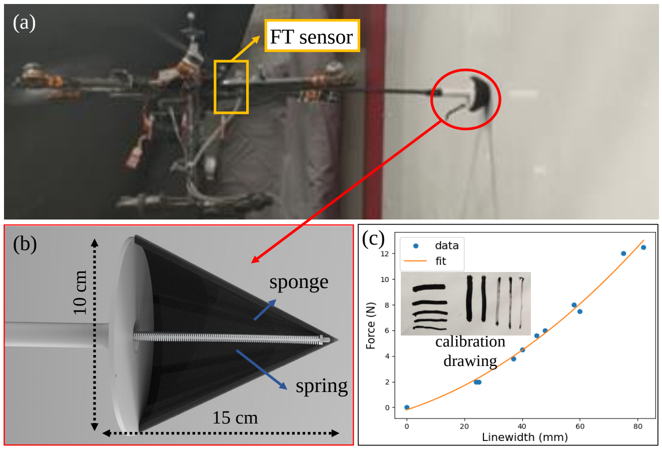

Fig. 4. System overview. (a). Side-view of our UAM platform. (b). Design of the sponge pen as the end-effector. (c). Calibrated stroke linewidth and contact force relation.

Notice that $\hat{\boldsymbol{\tau}}_c$ in (12), and $\hat{F}$ in (9) take the contact force, including the normal force, friction force and the induced torque into account. Therefore, the designed controller is a contact-aware hybrid motion-force controller.

VI. END-EFFECTOR AND INTERFACE DESIGNS

In this section, we introduce the end-effector and user interface design for the aerial calligraphy task.

A. End-Effector Design

Aerial calligraphy requires precise tracking of target linewidths, which can be achieved through contact force tracking. Most existing pens cannot produce varied linewidths, and specialized tools like the Chinese Calligraphy Brush need extra steps to align bristles for quality writing. In general, the pen for aerial calligraphy must satisfy these requirements: 1) Linewidth must correlate positively with contact force; 2) Changes in linewidth should be noticeable with varying force; 3) The correlation should be easily adjustable; 4) Writing should remain symmetrical during force application and release; 5) The pen should efficiently retain ink. We design a novel end-effector that meets these requirements.

Fig. 4 illustrates our pen design, consisting of two primary components: the sponge and the spring support. The cone-shaped sponge ensures isotropic properties and effectively absorbs and retains ink. A spring encircles a metal rod within the sponge, with adjustable stiffness and preload to fine-tune the force-linewidth curve. We dilute black ink with isopropyl alcohol (IPA) to enhance adhesion to the operating surface, which is a dry-erase whiteboard paper.

B. Aerial Calligraphy Interface

The interface allows users to input strokes from open-source datasets [29] or through manual design. We also develop a touchscreen interface, implemented as an iPad application, which enables direct drawing of target strokes by capturing

TABLE I
ABLATION STUDIES RESULTS

	Our Method	Control Baseline	Planning Baseline
EE Pos RMSE (cm)	$\mathbf{2.93 \pm 0.40}$	5.47 ± 0.74	4.40 ± 0.14
Base Pos RMSE (cm)	$\mathbf{2.48 \pm 0.39}$	4.07 ± 0.93	3.68 ± 0.65
Force RMSE (N)	$\mathbf{0.72 \pm 0.11}$	1.89 ± 0.07	1.04 ± 0.28
IoU	$\mathbf{0.59 \pm 0.09}$	0.43 ± 0.12	0.49 ± 0.06

waypoints and the pressure applied during drawing. For each user input target strokes s_i, including waypoints ${}_{\mathcal{I}}\boldsymbol{p}_i$ (position) in image space $\mathcal{I}$ and ${}_{\mathcal{I}}d_i$ (linewidth), the interface processes it using Cubicspline fitting with constraints such as minimal curvature, and maximum and minimum linewidth to ensure feasibility. The target contact force ${}^{r}F_i$ is then calculated from the pre-calibrated force-linewidth relationship. The position ${}^{r}\boldsymbol{p}_{ci}$ is obtained by transforming ${}_{\mathcal{I}}\boldsymbol{p}$ to frame $\mathcal{W}$. For multi-stroke letters, we add the non-contact point at the endpoints of each stroke to enable stroke switching. Subsequently, the developed contact-aware trajectory planning algorithm generates the reference motion-force trajectory $\boldsymbol{x}(t), \boldsymbol{\tau}_a(t)$, which the UAM tracks to draw the target letter. In the real realization, we run the optimization for each free flight period and continuous contact period and then stitch them together.

VII. EXPERIMENT

In this section, we conduct experiments to demonstrate the effectiveness of our method. We perform aerial calligraphy on several letters from different languages and investigate the impact of various system modules and writing speeds on overall performance. More writing results are available on our project website.

A. Experiment Setup

We use a customized fully-actuated hexarotor UAV with onboard computation handled by an Nvidia Xavier AGX running ROS Noetic. System localization is achieved using an Optitrack Motion capture system. For aerial calligraphy, the newly designed pen (Sec. VI-A) is mounted as the end-effector, and a dry-erase whiteboard paper is attached to a vertical wall for drawing. The friction coefficient is pre-measured as $\mu = 0.4$. The force-linewidth relationship is calibrated by drawing lines with varying contact forces and fitting a polynomial curve, as shown in Fig. 4. Trajectory planning and optimization is performed using CasADI [30] with IPOPT [31] solver, initialized with a 0.1s time interval and linearly interpolated states between the waypoints, and hover thrusts. In real applications, we control the vehicle base for implementation convenience.

B. Aerial Calligraphy

The aerial manipulator writes three English letters: ‘A’, ‘I’, and ‘R’ in a custom typeface and font size. We manually define the sparse waypoints with point locations and local linewidths. The maximum reference velocity is set to $20\ \mathrm{cm/s}$, the maximum acceleration to $20\ \mathrm{cm/s^2}$; and the maximum contact force change rate to $2\ \mathrm{N/s}$. We conducted three trials for each letter to test repeatability.

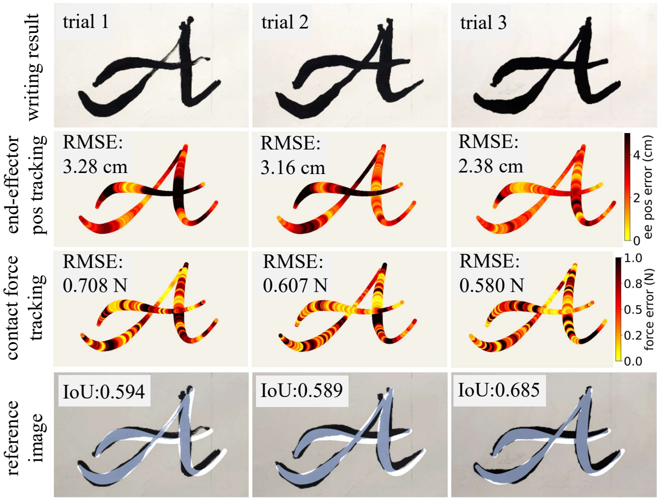

Fig. 5. Results of three repeated trials for the letter ‘A’. Top row: Final written image. Middle two rows: Time history of end-effector position and contact force tracking performance. Bottom row: visual comparison between the written letter and ground truth image, evaluating the overall writing performance.

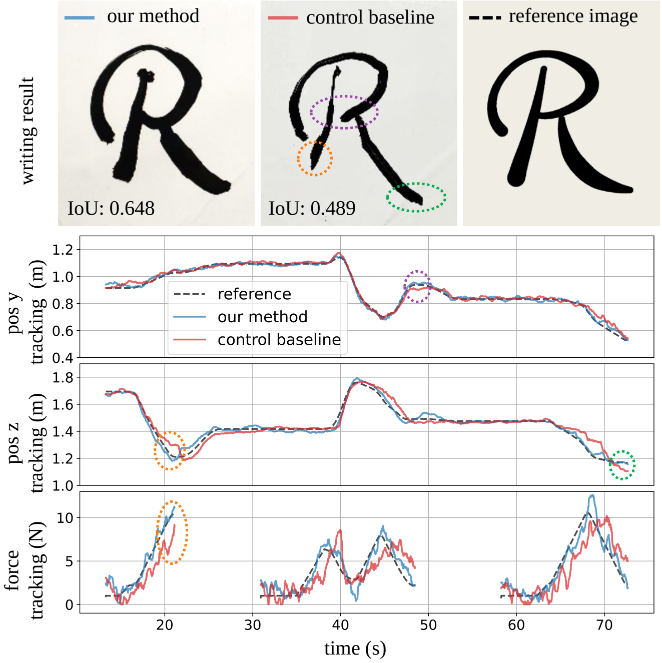

Fig. 6. Comparison with the control baseline. Top: Writing results for letter ‘R’. Bottom: Time history of end-effector motion and contact force tracking performance.

Fig. 5 shows the visual performance of the written letter. The UAM tracks motion and force trajectories well, with the written letters closely matching the target shapes and linewidths. Challenges arise when the UAM changes direction or at the start and end points of strokes due to the non-homogeneous property of the sponge pen during the

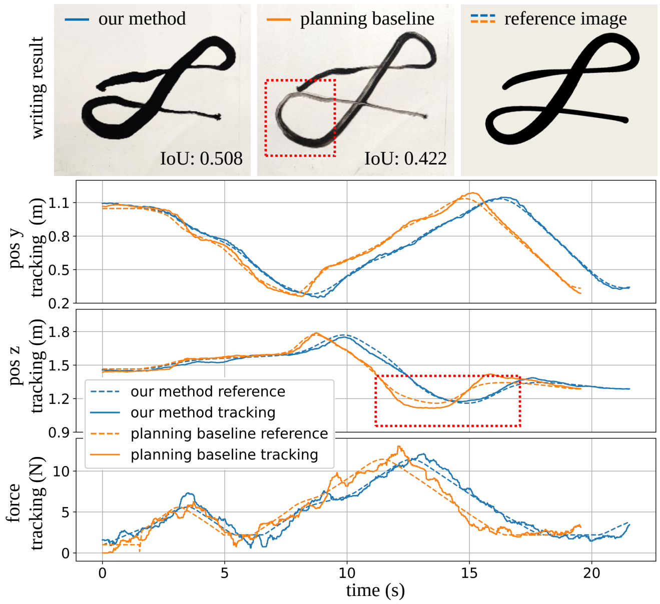

Fig. 7. Comparison with the planning baseline. Top: Writing results for letter ‘I’. Bottom: Time history of end-effector motion and contact force tracking performance.

establishment and detachment of contact. Smaller circular sections, such as the bottom-left curve of the letter ‘A’, or the bottom-left part of the letter ‘I’ (see Fig. 7), also present difficulties. This is presumably due to the rapid changing of friction mode (static or sliding friction) and friction force (both magnitude and direction) in these areas which requires a more accurate friction model. Despite these issues, the controller’s performance is consistent across trials.

To quantitatively evaluate the performance, we calculate the Root Mean Squared Error (RMSE) by comparing the reference trajectory, shown in Tab. I. The RMSE of the base position tracking is around 2.5 cm and of end-effector (ee) position tracking is around 2.9 cm. The contact force RMSE is around 0.7 N, which is better than the previous work that tracked a constant contact force [13]. This improvement is attributed to the developed contact-aware motion and force planning and control, which compensates for unmodeled behavior. We also use Intersection over Union (IoU) to evaluate the overall visual performance. We transform both the target and written letters into binary images, treating the letter as the target region at the pixel level. Our method achieves an average IoU of 0.59, with an average intersection ratio of 0.79 and an average union ratio of 1.35.

We also test our system on a more complex scenario: Chinese idiom calligraphy. As shown in Fig. 1, these characters feature more strokes, intricate shapes, and sharper turns. Our system still successfully wrote these characters with tracking and visual errors comparable to previous tests, highlighting the robustness of our approach across diverse writing tasks.

C. Ablation Studies

We conducted an ablation study to investigate the benefits of the proposed trajectory planning and hybrid motion-force control modules. We compared system performance under

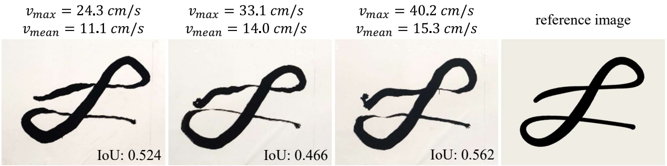

Fig. 8. Visual results of aerial writing on the letter ‘I’ under different velocity profiles.

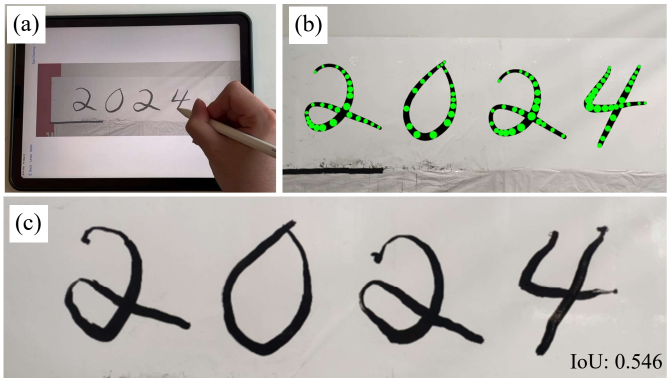

Fig. 9. (a). The user inputs target strokes ‘2024’ on the touchscreen. (b) The planned trajectory, where the green dots are the waypoints with equal timesteps. Regions with denser points indicate a slower planned velocity. (c). The actual letters written by the system.

three scenarios: 1) our proposed approach; 2) without the friction and contact force model in the controller (Control Baseline); and 3) without contact-aware trajectory planning (Planning Baseline), using equal distance sampling of target waypoints for reference trajectory instead, which only passes ${}^{r}\boldsymbol{q}$ into the controller. The results are shown in Tab. I.

Fig. 6 shows the comparison with scenario 2). It is clear that without a proper contact force model, the contact force trajectory tracking is less accurate, leading to inconsistent stroke widths and occasional loss of contact. Without a proper friction force model, the written letter shapes are distorted, particularly at the ends of strokes and junctions, such as the vertical line and junction of the letter ‘R’.

Fig. 7 shows the comparison with scenario 3). Without proper dynamic constraints, the linear interpolated reference trajectory makes sharp turns at corners, resulting in discontinuous velocity references that are unfeasible for the vehicle. This causes overshoots at corners, such as the bottom-left corner of the letter ‘I’. In addition, testing the Planning Baseline with a slower speed resulted in even worse performance due to frequent switching between static and sliding friction, introducing more complex friction dynamics.

D. Speed Comparison

To demonstrate the effects of the velocity on writing accuracy, we perform the experiments with three different velocity profiles whose maximum speed is 24.3 cm/s, 33.1 cm/s, and 40.2 cm/s. Fig. 8 shows the visual results. The proposed method achieves a final writing IoU of 0.562 at a velocity of 40 cm/s. To the best of our knowledge, this is significantly faster than all related works [22]. The main restriction on achieving higher velocities is the calibrated contact force change speed, limited by hardware and the contact model. Given that sliding at a slow speed introduces extra challenges, there is an optimal velocity range, highlighting the necessity of our trajectory planning algorithm.

E. Aerial Calligraphy with Touchscreen Interface

We demonstrate the convenience of our touchscreen interface for a complete aerial writing demo. A human user writes '2024' on the touchscreen. We then preprocess the input waypoints to make them feasible for our system. We regenerate the target image and render it for user verification. The UAM then performs contact-aware trajectory planning and hybrid motion-force control to draw the letters. Fig. 9 shows the user's raw input, the regenerated target image, and the written letters by the UAM. The touchscreen provides a more intuitive way for users to input targets.

VIII. CONCLUSION

In this letter, we propose a general framework to simultaneously track time-varying contact force in the surface's normal direction and motion trajectories on tangential surfaces. Specifically, we propose an aerial calligraphy system. Our pipeline imports predefined target strokes or user drawings from a touchscreen, generates a dynamically feasible trajectory, and controls the UAM to write the target calligraphy. Experiments demonstrate that our method can draw diverse letters with effective motion and force tracking. The IoU of the final drawing is 0.59, and the end-effector position tracking RMSE is 2.9 cm. Future work includes integrating the touchscreen as an online teleoperation interface.

ACKNOWLEDGMENT

This research is partially sponsored by the CMU Manufacturing Futures Institute (MFI). Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the MFI.

REFERENCES

[1] R. Bonatti, W. Wang, C. Ho, A. Ahuja, M. Gschwindt, E. Camci, E. Kayacan, S. Choudhury, and S. Scherer, "Autonomous aerial cinematography in unstructured environments with learned artistic decision-making," *Journal of Field Robotics*, vol. 37, no. 4, pp. 606–641, 2020.

[2] S. Zhao, H. Zhang, P. Wang, L. Nogueira, and S. Scherer, "Super odometry: Imu-centric lidar-visual-inertial estimator for challenging environments," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 8729–8736.

[3] J. Geng and J. W. Langelaan, "Cooperative transport of a slung load using load-leading control," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 7, pp. 1313–1331, 2020.

[4] ——, "Estimation of inertial properties for a multilift slung load," *Journal of Guidance, Control, and Dynamics*, vol. 44, no. 2, pp. 220–237, 2021.

[5] M. O'Connell, G. Shi, X. Shi, K. Azizzadenesheli, A. Anandkumar, Y. Yue, and S.-J. Chung, "Neural-fly enables rapid learning for agile flight in strong winds," *Science Robotics*, vol. 7, no. 66, p. eabm6597, 2022.

[6] G. Shi, X. Shi, M. O'Connell, R. Yu, K. Azizzadenesheli, A. Anandkumar, Y. Yue, and S.-J. Chung, "Neural lander: Stable drone landing control using learned dynamics," in *2019 international conference on robotics and automation (icra)*. IEEE, 2019, pp. 9784–9790.

[7] D. Lee, H. Seo, I. Jang, S. J. Lee, and H. J. Kim, "Aerial manipulator pushing a movable structure using a dob-based robust controller," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 723–730, 2020.

[8] M. Brunner, G. Rizzi, M. Studiger, R. Siegwart, and M. Tognon, "A planning-and-control framework for aerial manipulation of articulated objects," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 689–10 696, 2022.

[9] D. Lee, H. Seo, D. Kim, and H. J. Kim, "Aerial manipulation using model predictive control for opening a hinged door," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1237–1242.

[10] C. Ding, L. Lu, C. Wang, and C. Ding, "Design, sensing, and control of a novel uav platform for aerial drilling and screwing," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3176–3183, 2021.

[11] M. Schuster, D. Bernstein, P. Reck, S. Hamaza, and M. Beitelschmidt, "Automated aerial screwing with a fully actuated aerial manipulator," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 3340–3347.

[12] M. Alkaddour, M. A. Jaradat, S. Tellab, N. Sherif, M. Alvi, L. Romdhane, and K. S. Hatamleh, "Novel design of lightweight aerial manipulator for solar panel cleaning applications," *IEEE Access*, 2023.

[13] X. Guo, G. He, M. Mousaei, J. Geng, G. Shi, and S. Scherer, "Aerial interaction with tactile sensing," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 1576–1582.

[14] K. Bodie, M. Brunner, M. Pantic, S. Walser, P. Pfändler, U. Angst, R. Siegwart, and J. Nieto, "An omnidirectional aerial manipulation platform for contact-based inspection," *arXiv preprint arXiv:1905.03502*, 2019.

[15] J. Fishman and L. Carlone, "Control and trajectory optimization for soft aerial manipulation," in *2021 IEEE Aerospace Conference (50100)*. IEEE, 2021, pp. 1–17.

[16] M. Wang, Z. Chen, K. Guo, X. Yu, Y. Zhang, L. Guo, and W. Wang, "Millimeter-level pick and peg-in-hole task achieved by aerial manipulator," *IEEE Transactions on Robotics*, vol. 40, pp. 1242–1260, 2024.

[17] H. Lee, H. Kim, and H. J. Kim, "Planning and control for collision-free cooperative aerial transportation," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 1, pp. 189–201, 2016.

[18] M. Brunner, L. Giacomini, R. Siegwart, and M. Tognon, "Energy tank-based policies for robust aerial physical interaction with moving objects," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2054–2060.

[19] W. Zhang, L. Ott, M. Tognon, and R. Siegwart, "Learning variable impedance control for aerial sliding on uneven heterogeneous surfaces by proprioceptive and tactile sensing," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 275–11 282, 2022.

[20] L. Peric, M. Brunner, K. Bodie, M. Tognon, and R. Siegwart, "Direct force and pose nmpc with multiple interaction modes for aerial push-and-slide operations," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 131–137.

[21] K. Bodie, M. Brunner, M. Pantic, S. Walser, P. Pfändler, U. Angst, R. Siegwart, and J. Nieto, "Active interaction force control for contact-based inspection with a fully actuated aerial vehicle," *IEEE Transactions on Robotics*, vol. 37, no. 3, pp. 709–722, 2020.

[22] D. Tzoumanikas, F. Graule, Q. Yan, D. Shah, M. Popovic, and S. Leutenegger, "Aerial manipulation using hybrid force and position nmpc applied to aerial writing," *arXiv preprint arXiv:2006.02116*, 2020.

[23] C. Lanegger, M. Ruggia, M. Tognon, L. Ott, and R. Siegwart, "Aerial layouting: Design and control of a compliant and actuated end-effector for precise in-flight marking on ceilings," *Proceedings of Robotics: Science and System XVIII*, p. p073, 2022.

[24] J. Byun, J. Kim, D. Eom, D. Lee, C. Kim, and H. J. Kim, "Image-based time-varying contact force control of aerial manipulator using robust impedance filter," *IEEE Robotics and Automation Letters*, 2024.

[25] Z. Wang, X. Zhou, C. Xu, and F. Gao, "Geometrically constrained trajectory optimization for multicopters," *IEEE Transactions on Robotics*, vol. 38, no. 5, pp. 3259–3278, 2022.

[26] Y. Su, J. Li, Z. Jiao, M. Wang, C. Chu, H. Li, Y. Zhu, and H. Liu, "Sequential manipulation planning for over-actuated unmanned aerial manipulators," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 6905–6911.

[27] K. Lynch, *Modern Robotics*. Cambridge University Press, 2017.

[28] G. He, Y. Jangir, J. Geng, M. Mousaei, D. Bai, and S. Scherer, "Image-based visual servo control for aerial manipulation using a fully-actuated uav," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 5042–5049.

[29] S. Kishore, "Make me a hanzi," https://github.com/skishore/makemeahanzi, 2018.

[30] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, 2018.

[31] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, vol. 106, pp. 25–57, 2006.