Appendix I: Two potential modeling methods

In this section we delve deeper into the rationale for reformulating the frame-wise classification problem as a mixed-integer programming (MIP) under *Assumption 1* and *Assumption 2*, and for adopting the particular MIP structure used in the paper.

Under *Assumption 1*, only the intersection of *all* videos contains valid production actions, while *Assumption 2* stipulates that the action sequence be fixed. The task is therefore equivalent to locating, within the reference video, the subsequence that also appears in every other video.

There are *two* possible modeling routes, shown in Figures 15 and 16. Figure 15 corresponds to the formulation adopted in the paper: for each frame in video i we identify the most similar frame in the reference video, and only after all videos have been processed do we take the intersection to decide whether a reference action is truly indispensable. The assignment of frames within each video is thus *independent*. By contrast, Figure 16 starts from each reference frame k and asks whether that frame appears in *every* other videos.

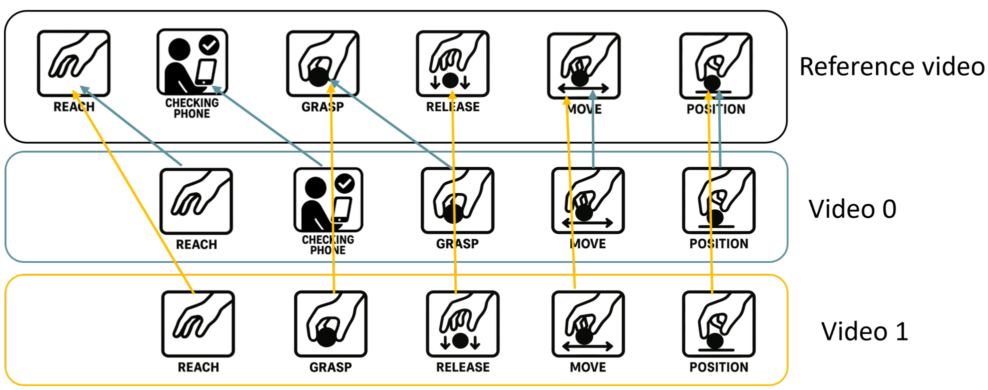


Figure 15 Locating, in the reference video, the subset of actions that appears in every production video (fixed step sequence). Each action in the other videos "votes" for its counterpart in the reference.

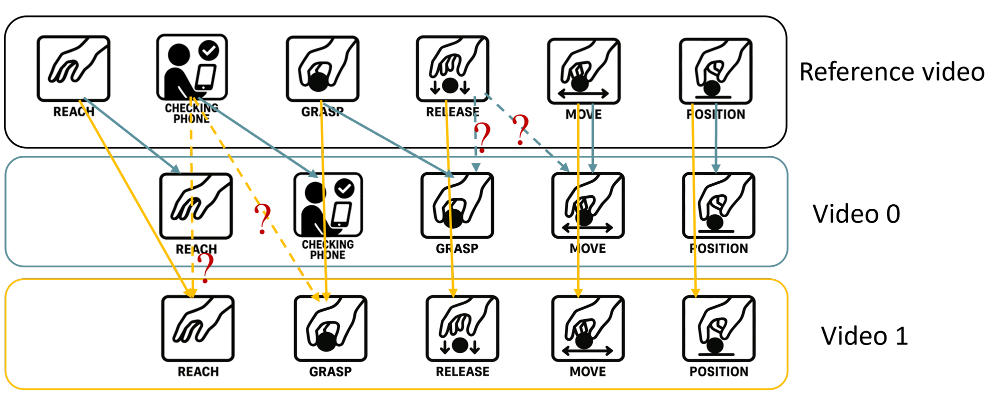


Figure 16 Alternative global formulation: each reference frame is directly checked against every production video.

Optimality. Because both formulations exploit exactly the same prior knowledge (Assumptions 1–2), the "no-free-lunch" principle implies that their expected error rates are identical once the assumptions are fully utilized.

Hyper-parameter simplicity. At first glance, Figure 16 appears more attractive because each decision "looks at" all videos simultaneously. In practice, however, one must define a distance threshold a such that $d_{k,t} < a$ is deemed a match—an approach reminiscent of control-chart techniques in anomaly detection. This threshold is highly scenario-dependent and undermines generalizability. It is not a likelihood ratio test or regularization term, but an absolute value threshold of whether a frame is matched or not. Under this modeling method we need to solve model (14).

$$\text{Minimise} \quad \sum_{k=0}^{l_0}\Big(\sum_{i=1}^{n}\sum_{t=0}^{l_i} d(\mathbf{z}_{i,t},\mathbf{z}_{0,k})\, x_{i,k,t} + (1-s_k)\, a\Big), \tag{14a}$$

$$\text{s.t.} \quad \sum_{t=0}^{l_i} x_{i,k,t} \le 1, \forall i \in \{1,\dots,n\},\ k \in \{0,\dots,l_0\}, \tag{14b}$$

$$\sum_{t=0}^{l_i}\sum_{k=0}^{l_0} x_{i,k,t} \ge n s_k, \tag{14c}$$

$$x_{i,k,t+1} \le 1 - x_{i,k',t}, \forall k \in \{0,\dots,l_0\},\ k' > k, \tag{14d}$$

$$x_{i,0,0} = 1, \forall i \in \{0,\dots,n\}, \tag{14e}$$

$$x_{i,l_i,l_0} = 1, \forall i \in \{0,\dots,n\}, \tag{14f}$$

$$x_{i,k,t} \in \{0,1\}, s_k \in \{0,1\}, \forall i \in \{1,\dots,n\},\ t \in \{0,\dots,l_i\},\ k \in \{0,\dots,l_0\}. \tag{14g}$$

Here $s_k = 1$ indicates that reference frame k represents a *necessary* action, whereas $x_{i,k,t} = 1$ records that frame t of video i is aligned to reference frame k. The cost in (14a) incurs $d(\mathbf{z}_{i,t},\mathbf{z}_{0,k})$ when a match is made, and a penalty a when the reference frame is skipped. Constraint (14c) says that a frame can be retained ($s_k = 1$) only if *every* video supplies at least one matching frame; together with (14b) this is equivalent to requiring n distinct matches in total. Constraint (14d) enforces the monotone ordering mandated by Assumption 2.

In contrast, the formulation in Figure 15 requires *no* such threshold: each frame in video i simply chooses its closest reference action. If a perfect match exists, its distance $d_{k,t}$ will indeed be minimal, so minimizing (14a) lets the correct alignment emerge automatically. The only additional parameter is the skip-penalty r_i to further improve robustness, which is interpretable via a likelihood-ratio argument. The formal mathematical formulation is shown in model (15) and the interpretation of the model can be found in the main text.

$$\text{Minimize} \quad \sum_{i=1}^{n}\Big(\sum_{t=0}^{l_i}\sum_{k=0}^{l_0} d(\mathbf{z}_{i,t},\mathbf{z}_{0,k})\, x_{i,t,k} + \sum_{k=0}^{l_0}\big(1-\sum_{t=0}^{l_i} x_{i,t,k}\big)^{+} r_i\Big), \tag{15a}$$

$$\text{s.t.} \quad \sum_{k=0}^{l_0} x_{i,t,k} = 1, \forall i \in \{1,\dots,n\},\ t \in \{0,\dots,l_i\}, \tag{15b}$$

$$x_{i,t+1,k} \le 1 - x_{i,t,k'}, \forall k \in \{0,\dots,l_0\},\ k' > k, \tag{15c}$$

$$x_{i,0,0} = 1, \forall i \in \{0,\dots,n\}, \tag{15d}$$

$$x_{i,l_i,l_0} = 1, \forall i \in \{0,\dots,n\}, \tag{15e}$$

$$x_{i,t,k} \in \{0,1\}, \forall i \in \{1,\dots,n\},\ t \in \{0,\dots,l_i\},\ k \in \{0,\dots,l_0\}. \tag{15f}$$

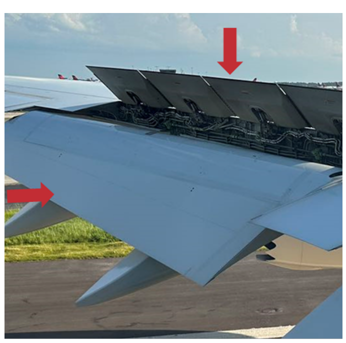
Computational complexity. Model (14) is a model similar to a *multiple alignment with SP-score* problem. *SP–score* ("sum–of–pairs score") is a classical objective in multiple–sequence alignment: one inserts gaps so that all m sequences have equal length, then sums the pairwise alignment costs over the $\binom{m}{2}$ sequence pairs. The optimization task is to find an alignment with the minimum SP–score and this problem has proved to be NP-complete (Elias 2006); obtaining an optimal solution at industrial scale is therefore infeasible. And we cannot transfer it into a DP formulation to solve it in polynomial time.We further validate this with numerical experiment on a computer with a CPU of i9 14900k. As shown in Table 8, with a reference sequence length of only $L = 50$, model (14) contains over 7.8k binary variables and the solver required 548 seconds without finding an optimal solution (the objective remained unresolved). In contrast, the model (15) formulation (our DP-based approach) remains tractable: By breaking the problem into independent alignments and exploiting domain assumptions, our method bypasses this complexity barrier. The DP-based formulation scales efficiently, providing exact solutions where the alternative formulation fails.

L_0	#Variables	Build Time (s)	Solve Time (s)	Objective
10	374	0.01	0.06	11.671587
20	1344	0.07	4.64	18.476656
50	7854	1.05	548.96+	Unsolved

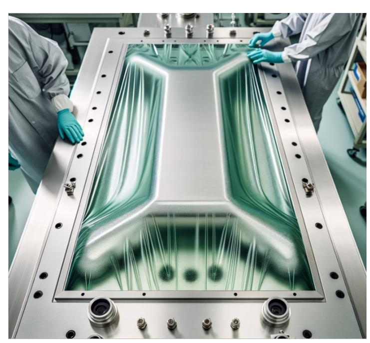
Table 8 **The alternative model is intractable with only 3 videos and each length of 50.**

Appendix J: Example of SOP generated by proposed method

In this section, we present a randomly selected example from the test production videos, showcasing the generated text description of standard operations along with the corresponding ground truth for each dataset. It can be seen that most action descriptions are accurate; however, some specific terms for production parts or tools were not correctly matched. As large vision-language models continue to improve in their captioning capabilities, especially with the development of models tailored for industrial datasets, the performance of our algorithm is expected to improve significantly as well.



(a)



(b)

Figure 17 **(a) Example of commercial aircraft spoiler, (b) AI-generated spoiler production sketch map.**

J.1. Aircraft Spoiler Production

Step Name	Procedure	Predicted Step Name	Predicted Procedure
Applying composite material.	The worker places the yellow composite material on the tooling. And the yellow layer is removed with **red material** remain and wait for dry out.	Applying the **red paint** to the spoiler.	The worker in the video is responsible for applying a substance to the spoiler, which is a part of the airplane wing. The substance is applied to the spoiler to ensure that it is properly coated and ready for use. **The worker carefully applies the substance to the spoiler**, ensuring that it is evenly distributed and free of any bubbles or imperfections. The worker then uses a **tool to smooth out the substance**tool to smooth out the substance, ensuring that it is applied correctly and that the spoiler is ready for use.
Insert rigid structure.	The worker stick the yellow rigid structure on the red material and **cover it with red cloth** carefully. After that, he **insert the green combined rigid structure**.	The step involves cleaning the spoilers.	The worker in the video is responsible for cleaning the spoiler of an airplane. He starts by wiping the spoiler **with a cloth**, then applies a black substance to the spoiler and wipes it off. Finally, he **applies a green substance** to the spoiler and **wipes it off.**wipes it off.
Apply compensation tape	**The worker wipe the combined rigid structure**The worker wipe the combined rigid structure, **pushes it down to make sure the structure is stuck in the tooling**pushes it down to make sure the structure is stuck in the tooling and apply green compensation tape, this step should be performed under the collaboration of multiple workers.	Assemble and test.	The worker in the video is responsible for assembling the spoilers. **He starts by placing the spoiler on a table and then proceeds to install the spoiler**He starts by placing the spoiler on a table and then proceeds to install the spoiler on the airplane. He then installs the spoiler on the table and moves it to the airplane. The worker also installs the spoiler on the table and moves it to the airplane.
Assembling mold and inspecting.	The worker is responsible for assembling and inspecting the spoilers. Starting by **placing the metal mold on the table and then proceeds to secure it in place**placing the metal mold on the table and then proceeds to secure it in place. Then moves on to the inspecting phase, where the worker checks the spoiler's functionality and ensures that it is working correctly.	Cleaning the spoiler.	The worker in the video is responsible for cleaning the spoiler of an airplane. He starts by cleaning the spoiler with a cloth and then proceeds to clean the spoiler with a brush. After that, he cleans the spoiler with a sponge and then wipes it down with a cloth. Finally, he cleans the spoiler with a brush again.

Seal up	The worker starts by **cover the spoiler with a red transparent plastic film**cover the spoiler with a red transparent plastic film, then uses a tool to scrape off the excess part.	**Wrapping the spoilers in plastic.**	The worker in the video is responsible for **applying a substance to the spoiler**applying a substance to the spoiler, which is a part of an airplane wing. He uses a brush to apply the substance evenly on the spoiler, **ensuring that the entire surface is covered.**ensuring that the entire surface is covered. The purpose of this operation is to improve the aerodynamics of the spoiler, which helps in reducing air resistance and increasing the overall performance of the airplane.
Seal up	The worker uses blue tape to seal up the plastic film.	\	\
Process cover plate	The worker should starts by **placing the process cover plate on the spoiler** and then proceeds to connect the wires to the spoiler.	Cleaning and polishing the spoiler.	The worker in the video is responsible for cleaning the spoiler, which is a part of the airplane wing. He uses a cloth to wipe the spoiler clean and **then applies a substance to it.** The substance is likely a cleaning solution or a **protective coating.** The worker then wipes the spoiler again to ensure that it is clean and free of any debris or contaminants. This process is crucial to ensure the spoiler's proper functioning and to maintain the airplane's overall performance.
Apply white cloth	The worker is responsible for covering the airplane spoiler with a white cloth and **scrape off the excess part of** it and make sure the cloth cover all the spoiler.	**Wrapping the spoiler.**	The worker in the video is responsible for the production of airplane spoilers. He is seen using a tool to **cut a piece of paper**, which is then placed on a table. He then uses a vacuum to clean the table, ensuring that it is free of any debris.
Bagging.	The worker is responsible for **covering the airplane spoiler with a plastic sheet. Workers should carefully wraps the sheet around the spoiler and ensures that it is securely in place. Then tapes the plastic sheet to the spoiler.**	**Wrapping the spoiler in plastic.**	The worker in the video is responsible for **covering the airplane spoiler with a plastic sheet. He starts by cutting the plastic sheet to the appropriate size, then places it over the airplane spoiler. He then uses a tool to press the plastic sheet down onto the spoiler, ensuring that it is securely in place.**
Vacuum.	The worker should connect the vacuum pump to pump out the air under the plastic film. Finally, **he press the plastic down again to ensure that all the air can be evacuated by the vacuum pump and the plastic film is tightly attached to the workbench.**	Wrapping the spoilers.	The worker in the video is seen **covering the airplane spoiler with a plastic sheet. He then uses a tool to press the plastic sheet onto the spoiler, ensuring that it is securely in place.** This is a crucial step in the production process, as it helps to protect the spoiler from dust and debris while it is being transported or stored. The worker's attention to detail and precision in this operation is essential to ensure that the spoiler is properly protected and ready for use.

J.2. Water Valve Production

Step Name	**Procedure**	**Predicted Step Name**	**Predicted Procedure**

Inserting rubber stopper	The left hand holds the object1, while the right hand rotates the object1 and insert the rubber stopper. The left hand then releases the object1 and flip it, and the right hand continues to rotate the object1.	Tightening the cap	The left hand holds the object, while the right hand rotates it. The left hand then releases the object, and the right hand picks it up. The left hand then rotates the object again, and the right hand picks it up again. This process is repeated a few times.
Assembling inter-mediate valve	The left hand holds the object1, while the right hand insert the intermediate valve body, and the right hand flip the assembled parts.	Assembly step 1: Install the water valve.	The left hand holds the object, while the right hand rotates it. The left hand then releases the object, and the right hand rotates it again. This process is repeated several times.
		Assembly step 1: Attach the metal pipe to the valve body.	The left hand holds the container and the right hand rotates the container. The left hand then releases the container and the right hand rotates the container again. The left hand then rotates the container again and the right hand rotates the container.
Fastening inter-mediate valve	The left hand holds the object, while the right hand fastening the screw.	Assembly step 1: Attach the water valve to the pipe.	The left hand unscrews the top of the object, and the right hand inserts a metal pipe into the hole. The left hand then screws the top back on.
Aseemble the lower valve	The left hand holds the assembled intermediate valve body and inserts it into the lower part of the valve.		
Aseemble the middle valve	The left hand holds the assembled workpieces and the right hand put the sealing ring and middle valve body on it.	Tightening the nut	The left hand unscrews the lid of the coffee press, and the right hand inserts a metal rod into the coffee press. The left hand then screws the lid back on and tightens it with the right hand. The right hand then rotates the metal rod to press the coffee grounds.
Insert sealing element	The left hand and right hand inserts the sealing element into the hole. The left hand then pull the metal part back in place.		
Insert copper head	The right hand insert the copper head.		
Attach the handle to the valve	The left hand holds the valve, while the right hand rotates the handle to tighten the screw.	Assembly step 1.	The left hand holds the handle of the tool, while the right hand rotates the handle to tighten the screw. The left hand then rotates the handle to loosen the screw.

Attach the handle to the valve	The left hand is responsible for turning the knob, while the right hand is responsible for pulling the handle. The left hand then releases the handle, and the right hand pulls the handle again. This process is repeated several times.		
Fastening the handle	The right hand holds the handle of the tool while the right left tighten the screw.		
Fastening the handle	The left hand holds the valve and the right hand rotates the screw driver to fasten the screw on the handle. This process is repeated until the screw is tightened to the desired level.	Tightening the nut	The left hand is shown holding a tool, while the right hand is shown holding a screw. The left hand then proceeds to insert the tool into the screw, and the right hand rotates the screw. The left hand then removes the tool and the right hand rotates the screw again. This process is repeated a few times, and the left hand then inserts the tool again and rotates the screw.
Fastening the handle	The left hand is responsible for holding the tool, while the right hand is responsible for adjusting the screw. The left hand then rotates the tool, while the right hand uses the tool to tighten the screw. The left hand then rotates the tool again, and the right hand uses the tool to tighten the screw. This process is repeated until the screw is tightened to the desired level.	Tightening the nut	The left hand holds the vice grip and the right hand holds the vice grip tool. The left hand then rotates the vice grip tool to tighten the vice grip. The right hand then rotates the vice grip tool to loosen the vice grip.
Cover the plastic cover	The right hand cover the plastic cover on the handle.	Assembly step 1.	The left hand is responsible for adjusting the pipe, while the right hand is responsible for tightening the pipe. The left hand rotates the pipe to the desired position, and then the right hand tightens the pipe using the wrench.
Attach clamp to the lower valve	The left hand and right hand clamp the clamp onto the lower valve body and middle valve body, and tighten the screw of the clamp with the right hand.	Assembly step 1: Attach the handle to the valve.	The left hand holds the metal piece, while the right hand rotates the metal piece. The left hand then inserts the metal piece into the hole, and the right hand rotates the metal piece to tighten it.
Attach clamp to the upper valve	The left hand and right hand clamp the clamp onto the upper valve body and middle valve body, and tighten the screw of the clamp with the right hand.	Assembly step 1.	The left hand holds the pipe and rotates it, while the right hand rotates the pipe with the left hand. The left hand then rotates the pipe with the right hand.

		Assembly step 1: Attach the valve body to the pipe.	The left hand holds the yellow object, which is a paper towel, and the right hand holds the machine. The left hand then wipes the machine with the paper towel, while the right hand rotates the machine. The left hand then wipes the machine again with the paper towel.
		Attach the valve to the pipe.	The left hand holds the paper, while the right hand rotates the object. The left hand then moves the paper to the right hand, and the right hand rotates the object again. The left hand moves the paper to the right hand, and the right hand rotates the object one more time. The left hand moves the paper to the right hand, and the right hand rotates the object one final time.

Appendix K: Pseudo codes

K.1. DSF pseudo code

Algorithm 1 Dynamic Similarity Filtering (DSF)

```
function DSF(z_{i,t})
    Initialize dp[k,t] ← ∞ for all k,t
    dp[0,0] ← 0
    for each i ∈ {1,...,n} do
        for each k ∈ {0,...,l_0} do
            for each t ∈ {0,...,l_i} do
                Compute cost ← d(z_{i,t}, z_{0,k})
                dp[k,t] ← min(dp[k−1,t] + r_i, dp[k,t−1] + cost, dp[k−1,t−1] + cost)
            end for
        end for
    end for
    Backtrack to determine x_{i,t,k}
    for each k ∈ {0,...,l_0} do
        if ∑_{t=1}^{l_i} x_{i,t,k} = 0 for any i then
            z_{0,k} is Redundant
        else
            z_{0,k} is Necessary
        end if
    end for
    Refine index of clusters, eliminating redundant clusters
    return DSF, p_k
end function
```

K.2. CECD pseudo code

Algorithm 2 Cosine-Energy Change-point Detection

Require: A sequence of feature vectors $\{\mathbf{z}_k\}_{k=1}^{l}$

```
function StepBoundaryDetection({z_k}_{k=1}^l, T = 2N)
    Initialize a zero vector d with length l
    for k_c = N + 1 to l − N do
        Compute D̂(k_c) for the window [k_c − N, k_c + N] according to Eqn. (7)
        Set d_{k_c} = D̂(k_c)
    end for
    Initialize an empty list of change-points C
    for k_c = N + 1 to T − N do
        if d*_{k_c} is the local maximum within window [k_c* − N, k_c* + N] then
            Perform a permutation test to compute the p-value for k_c*
            if p-value < α then
                Add k_c* to the list of change-points C
            end if
        end if
    end for
    return C
end function
```