MINI PROJECT

OPTIMIZATION OF DRYING PROCESS PARAMETERS

USING MACHINE LEARNING

SUBMITTED BY:

AATISH KHADRIA (2023UCH1197)

ROMIL YADAV (2023UCH1156)

ADITYA (2023UCH1181)

GORISHANKER PRAJAPAT (2023UCH1160)

SUBMITTED TO:

PROF. KAILASH SINGH

DATE OF SUBMISSION: 15/NOVEMBER/2025

1. REPORT SUMMARY:

This research presents a combined convection-infrared hot-air dryer that incorporates a computer-vision system for real-time monitoring of kiwifruit drying. Various drying parameters, including air temperature, air velocity, and infrared energy, were evaluated to determine their impact on drying efficiency and the quality of the fruit. Image processing methods were employed to assess changes in color and size without harming the fruit. The researchers utilized Response Surface Methodology (RSM) to model the drying process and establish the most effective operational parameters. The optimized system achieved quicker drying times, reduced energy consumption, and improved retention of fruit quality. Experimental results validated the model's predictions, demonstrating the system's dependability.

KEY FINDINGS OF THE REPORT:

- Increased temperature and infrared energy significantly shortened drying duration.

- Nevertheless, elevated settings led to greater colour darkening and increased firmness of the fruit.

- The identified optimal conditions included a temperature of 62.75 °C, infrared power of 248.28 W, and an air speed of 1 m/s.

- Under these conditions:

- Drying duration = 64.87 minutes

- Energy usage = 27.70 kWh/kg

- Energy efficiency = 16.92%

- Colour change and shrinkage were kept to a minimum.

- Experimental findings closely aligned with the model, indicating high reliability.

- The combination of computer vision and RSM optimization results in an intelligent, efficient dryer that maintains quality, making it ideal for contemporary fruit-processing operations.

2. Problem Statement:

The drying of high-moisture fruits including kiwifruit regularly outcomes in lengthy processing times, high energy consumption, and substantial pleasant deterioration while drying parameters aren't well optimized. The hybrid convection–infrared drying system used inside the take a look at includes more than one interacting variables—air temperature, air velocity, and infrared strength, every influencing drying kinetics, energy conduct, and product high-quality in a nonlinear manner. conventional methods fail to concurrently optimize those conflicting objectives or offer actual-time, non-unfavourable monitoring of best changes including shade and shrinkage.

So consequently, there may be a want to broaden a predictive and optimization framework that integrates system-getting to know ML modelling to decide the optimal aggregate of drying parameters that minimize drying time and electricity consumption at the same time as maintaining product first-rate.

3. Introduction :

Drying is one of the most widely used food preservation methods for extending shelf life and reducing post-harvest losses. Kiwifruit, being a high-moisture and nutrient-rich fruit, deteriorates rapidly unless properly dried. Conventional hot-air drying often suffers from long processing time, uneven heat transfer, and poor quality retention. Infrared (IR)-assisted drying has emerged as an efficient hybrid method, capable of improving heat transfer and reducing energy usage. However, drying performance is highly dependent on control of key parameters such as temperature, airflow, and IR power. Recent work highlights the potential of machine learning and statistical optimization techniques like Response Surface Methodology (RSM) to predict moisture removal, energy consumption, and product quality. At the same time, computer vision has proven to be a powerful, non-destructive tool for monitoring colour and dimensional changes during drying. Yet, a combined framework that integrates ML/RSM modelling with image-based quality monitoring for multi-objective optimization of drying is still lacking. This project utilizes the kiwifruit drying study to address this gap by analysing how airflow, temperature, and infrared power affect drying kinetics, energy behaviour, and quality attributes, and how machine learning approaches can be used to optimize them.

conventional warm-air drying has been used for decades, but it suffers from several drawbacks including long processing instances, uneven heat distribution, high energy call for, and tremendous nice loss in phrases of coloration, texture, and shrinkage. those boundaries have pushed the meals processing industry in the direction of hybrid technologies along with infrared (IR)-assisted convective drying. IR heating complements internal moisture diffusion and accelerates evaporation, resulting in shorter drying times and potentially decrease energy consumption. but, these advantages are finished best when vital parameters—which includes air temperature, air pace, and IR electricity—are cautiously optimized. Poorly decided on settings can accelerate high-quality degradation, boom power wastage, and result in inconsistent drying behaviour.

4. Objectives :

I. **To analyse the effect of drying parameters** (air temperature, air velocity, and IR power) on drying time, energy consumption, and energy efficiency.
II. **To evaluate fruit quality changes** (colour change, shrinkage, and firmness) using a non-destructive computer vision system.
III. **To develop predictive models** for drying responses using machine learning **.**
IV. **To perform multi-objective optimization** to find the best combination of drying parameters that minimize:

- drying time
- energy consumption
- quality degradation (colour change, shrinkage, firmness)
  while maximizing energy efficiency

V. **To validate the optimized conditions experimentally** and compare them with model predictions.

5. <u>Theoritical Background</u> :

4.1 Drying of Agricultural Products :

Drying involves simultaneous heat and mass transfer, where moisture inside the product moves to the surface and evaporates.
Key factors influencing drying:

- Air temperature
- Air velocity
- IR radiation power
  Higher temperature and IR power generally increase drying rate, but may also cause surface hardening, colour degradation, and shrinkage.

4.2 Infrared-Assisted Hot-Air Drying :

IR provides radiation heating directly to the fruit surface, improving efficiency compared to convection alone.
Benefits:

- Faster moisture removal
- Lower energy consumption
- Better quality retention (when controlled properly)
  Drawback:
- Higher IR power may increase browning and firmness.

4.3 Computer Vision for Real-Time Quality Assessment :

Computer vision measures:

- Colour change (ΔE) via RGB → L*a*b* conversion
- Shrinkage via segmentation and area reduction
- Shape and structural changes
  It is non-destructive, fast, and adaptable for real-time monitoring.

4.4 Energy Metrics in Drying :

Important indices analysed:

- Specific Energy Consumption (SEC): Energy used per kg of evaporated water
- Energy Efficiency (η): Fraction of supplied energy actually used to evaporate moisture
  High SEC and low efficiency indicate poor process design.

6. RESULTS AND DISCUSSION

7.1 Effect of Drying Parameters on Drying Time, Energy Consumption, and Energy Efficiency

The experimental results showed that air temperature, air velocity, and IR power had significant effects on the drying kinetics. Increasing air temperature from low to high levels markedly reduced drying time because higher temperatures increased the vapor pressure deficit and enhanced moisture diffusion. Similarly, higher IR power accelerated surface water evaporation, particularly during the initial stages of drying. However, increased air velocity contributed only moderately to moisture removal, mainly by reducing the boundary-layer resistance.

7.2 Development of Predictive Machine Learning Models

Machine learning models were trained to predict drying time, energy usage, and quality parameters based on input variables (temperature, velocity, IR power). Among the algorithms tested, Random Forest and Gradient Boosting showed the highest accuracy, with R^2 values above 0.90 for most responses. These tree-based models successfully captured nonlinear interactions between parameters.

Model validation results indicated that the ML predictions closely matched experimental observations, demonstrating their ability to generalize to unseen conditions. The models provided a reliable framework for process optimization without requiring extensive physical experiments.

7.3 Experimental Validation of Optimized Conditions

The optimized drying conditions obtained from the multi-objective model were tested experimentally. The results confirmed that the selected parameters achieved:

- Reduced drying time compared to baseline conditions
- Lower specific energy consumption
- Improved quality retention (colour, shrinkage, firmness)
- Higher energy efficiency

Model predictions showed strong agreement with experimental data, with deviations generally below 5–7%. This validates the effectiveness of ML-based optimization in predicting real drying behaviour.

7. Conclusion

This study successfully investigated and optimized the drying process by integrating experimental analysis, computer-vision–based quality evaluation, machine learning modelling, and multi-objective optimization techniques. The results demonstrated that drying parameters—particularly air temperature, air velocity, and IR power—strongly influence drying kinetics, energy consumption, and product quality. Higher temperatures and IR power significantly reduced drying time but also increased the risk of quality degradation, emphasizing the need for balanced operating conditions.

The non-destructive computer vision system proved effective for monitoring fruit quality changes such as color, shrinkage, and firmness, providing rapid and reliable assessment throughout the drying process. Machine learning models developed during the study showed high predictive accuracy for drying responses, enabling precise estimation of drying time, energy usage, and quality attributes without the need for repetitive experimentation

8. References

1. **GOOGLE COLAB**

   https://colab.research.google.com/drive/1xovya3dzHn40ija2VaPnVy4hCek9Jo7b?usp=sharing

2. **DATASET**

   https://1drv.ms/x/c/8af9bd2ca5dde2b2/EYxKqUdv321PqyxCvW5INWsBVI31SNLS0eFdnSLIjx0i6g?e=hXHBQI

3. **DATASET REFERENCE**

   file:///C:/Users/Aditya/OneDrive/%E3%83%89%E3%82%AD%E3%83%A5%E3%83%A1%E3%83%B3%E3%83%88/projectaiml.pdf

9. Appendix

1. CODE

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler


df = pd.read_csv('/content/drying_experiment_data.csv')


X = df[['Air temperature (°C)', 'IR Power (W)', 'Air velocity
(m/s)']]
y = df[['Drying time (min)', 'Specific energy (kWh/kg water)',
'Energy efficiency (%)', 'ΔE', 'Shrinkage (%)', 'Firmness (N)']]


X_train, X_test, y_train, y_test = train_test_split(X, y,
                                                    test_size=0.2,
random_state=42)


scaler = StandardScaler().fit(X_train)
X_train_scaled = scaler.transform(X_train)
X_test_scaled = scaler.transform(X_test)

from sklearn.neural_network import MLPRegressor

mlp = MLPRegressor(
    hidden_layer_sizes=(64, 32),  # two hidden layers
    activation='relu',
    solver='adam',
    max_iter=1000,
    random_state=42
)

mlp.fit(X_train_scaled, y_train)


from sklearn.metrics import r2_score

# NOTE: If you used hyperparameter search you should use the
selected model variable.
# The name 'best_mlp' used below will raise a NameError unless
defined. Use 'mlp' if this is the trained model.
y_pred_best = mlp.predict(X_test_scaled)

print("R² scores for the best MLP model on the test set:")
for i, col in enumerate(y.columns):
    r2 = r2_score(y_test[col], y_pred_best[:, i])
    print(f"{col:35s}: R² = {r2:.3f}")
```

```python
from sklearn.neural_network import MLPRegressor


mlp = MLPRegressor(
    hidden_layer_sizes=(64, 32),
    solver='adam',
    max_iter=1000,
    random_state=42
)
mlp.fit(X_train_scaled, y_train)

new_entry = pd.DataFrame({
    'Air temperature (°C)': [65],
    'IR Power (W)': [275],
    'Air velocity (m/s)': [1.75]
})


new_entry_scaled = scaler.transform(new_entry)


predicted_outcomes = mlp.predict(new_entry_scaled)


print("Predicted Outcomes for the new entry:")
for i, col in enumerate(y.columns):
    print(f"{col:35s}: {predicted_outcomes[0][i]:.3f}")
```